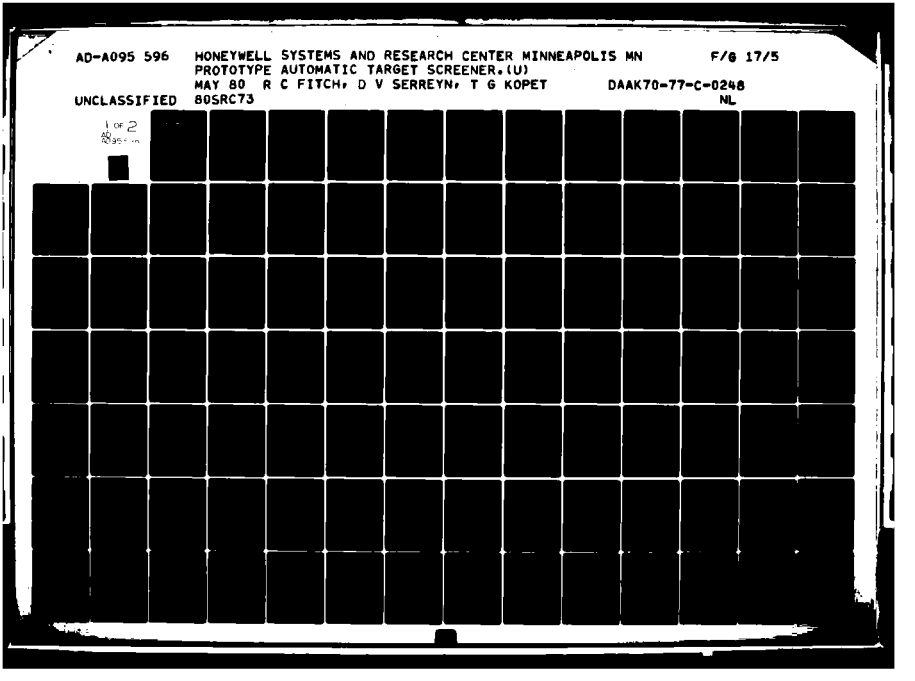
AD-A095 596 HONEYWELL SYSTEMS AND RESEARCH CENTER MINNEAPOLIS MN F/G 17/5
PROTOTYPE AUTOMATIC TARGET SCREENER.(U)
MAY 80 R C FITCH, D V SERREYN, T G KOPET DAAK70-77-C-0248
UNCLASSIFIED 80SRC73 NL

1 OF 2
AD
A095-596



PROTOTYPE AUTOMATIC TARGET SCREENER

by

R. C. Fitch
T. G. Kopet
D. V. Serreyn
R. C. Reitan
M. O. Schroeder

19 May 1980

REPORT FOR PERIOD

1 July 1979 – 31 December 1979

NIGHT VISION AND ELECTRO-OPTICS LABORATORY
FORT BELVOIR, VIRGINIA 22060

HONEYWELL
SYSTEMS & RESEARCH CENTER
2600 RIDGWAY PARKWAY
MINNEAPOLIS, MINNESOTA 55413

"The views, opinions, and/or findings contained in this report are those of the authors and should not be construed as an official department of the Army position, policy, or decision, unless so designated by other documentations."

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A095 596 / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PROTOTYPE AUTOMATIC TARGET SCREENER	5. TYPE OF REPORT & PERIOD COVERED Quarterly Progress Report No. 8, 1 July - 31 December 1979
	6. PERFORMING ORG. REPORT NUMBER 80SRC73
7. AUTHOR(s) R. C. Fitch, D. V. Serreyn, T. G. Kopet, R. C. Reitan, M. O. Schroeder	8. CONTRACT OR GRANT NUMBER(s) DAAK70-77-C-0248
9. PERFORMING ORGANIZATION NAME AND ADDRESS Honeywell Systems and Research Center, 2600 Ridgway Parkway, Minneapolis, MN 55413	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 1 E263710DK70 14 010CJ
11. CONTROLLING OFFICE NAME AND ADDRESS Night Vision and Electro-Optics Laboratory, Fort Belvoir, Virginia 22060	12. REPORT DATE 19 May 1980
	13. NUMBER OF PAGES 110
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Infrared	Target recognition	Image enhancement
FLIR	Pattern recognition	
Target cueing	Image processing	
Target screening	Real time	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report is the eighth quarterly progress report for contract DAAK70-77-C-0248, Prototype Automatic Target Screener. The objective of the effort is to design an automatic target screener to be used with thermal imaging systems employing common module components.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

CONTENTS (concluded)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (concluded)

LIST OF TABLES

SECTION 1

INTRODUCTION AND SUMMARY

This is the eighth technical progress report for contract DAAK70-77-C-0248, Prototype Automatic Target Screener (PATS). The first two reports document the Phase I design study. The third report provides a description of the final target classifier design for the target base currently available and the results of the hardware and CPU1 software system design tasks. The fourth through seventh reports describe further the subsystem design details and the status of the fabrication, software coding, and hardware checkout. This report covers hardware modifications we have made since the last report, the modification to include range target/clutter discrimination, and the general status of the PATS system. This report covers the period from July 1 to December 31, 1979.

The program objective is to produce a design for an automatic target screener. The screener will reduce the task loading on the thermal imager operator by detecting and recognizing a limited set of high-priority targets at ranges comparable to or greater than those for an unassisted observer. A second objective is to provide enhancement of the video presentation to the operator. The image enhancement includes: 1) automatic gain/brightness control to relieve the operator of the necessity of continually adjusting the display gain and brightness controls and 2) DC restoration to eliminate artifacts resulting from AC coupling of the infrared (IR) detectors.

Image enhancement will also include local area gain and brightness control to enhance variations of contrast and compress the overall scene dynamic range to match that of the display. This circuitry has been completed, and examples of its performance on videotaped thermal image data were included in the first quarterly report, along with the circuit description.

The DC restoration image enhancement circuit eliminates the streaking associated with loss of line-to-line correlation on the displayed image because of the AC coupling of the detector channels.

The program objective has been expanded to include flyable hardware and to have the LOHTAD laser range finder provide range inputs. Range input can also come from a transponder or be input manually.

This report consists of six Sections. The first section is this introduction and summary. Section 2 covers the hardware status and design modifications including capability for laser range finder inputs. Section 3 discusses the software status and describes the specific program developed in CPU2 as design and checkout aids. Section 4 contains plans for the next period prior to the final report. Appendix A is a preliminary test plan as preparation for flight and ground tests. Appendix B is a copy of a paper presented at the Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance Workshop held 19-20 November 1979 at USA MICOM, Redstone Arsenal, Alabama.

SECTION 2

HARDWARE DESIGN AND STATUS

This section describes the hardware status, a modification to the interval design, modification to the global gain/bias control circuitry, and the changes or hardware added to make the system flyable. These items are the CPU1 to symbol interface, the range interface, and a test box.

STATUS OF MODULES

Table 1 presents the modules that are in the PATS system. All the modules listed make the system a stand-alone piece of hardware. A test box external to the main box is required, however, for range input from a transponder.

All the boards listed are functioning. Because of noise problems encountered at the high clock rate, the physical layout of the boards may have changed a little but functionality is still present for all boards.

In addition to those modules listed in Table 1, an LSI 11/2 microcomputer (CPU2) is required for doing the training and diagnostics. Two boards that were not supplied with the computer were the DMA-L11 board described in the 6 April 1979 quarterly report and the CPU2 to symbol interface also described in the 6 April report. To the LSI 11/2 we have added an additional 16K words of memory to enable us to do the sorting of the K-nearest neighbor feature data.

TABLE 1. PATS MODULES

Function	Slot Assigned	Applicable Schematics
Adaptive contrast enhancement CCD delay line #5	A1	10069490 10069486
DC restore and global gain/bias CCD delay line #1	A2	10069840 10067922
Sync separator and video switch	A4	10069841
Sync Generation	A6	10069484
Bright CCD delay line #4	A8	10069060 10068106
Edge CCD delay line #2 CCD delay line #3	A10	10069069 10068104 10068105
Two H analog delay	A11	10077255
First level features	A12	10070193
Power monitor and CPU1/symbol	A14	10070196
Interval generation	A18	10070192
Memory #2	A21-A24	10067925
A/D, summation, background est	A26	10069253
Memory control and refresh	A27	10069251
Symbol generation	A28	10070190
CPU1/CPU2 interface and micro program memory	A29	10070195
FIFO/DMA interface	A31	10070188
Microprogram; memory sequencer	A32	10070187
CPU1 and multiplier	A34	10070185
Memory #1 constants	A36	10070189
Memory #1	A37	10070186

The writable control store is used for microprogram development and becomes a part of the Intel MDS system. The testing box includes a board for range input and is described elsewhere in this report.

INTERVAL DESIGN MODIFICATION

The PATS interval generation/validation board has been checked out. All of the bidirectional turn-on/off criteria are working, but some changes had to be made to the algorithm described in the previous PATS quarterly report. These changes as well as other checkout results are documented here. Specifically, we know that the signal timing and the nature of the interval signal will introduce offset error which must be corrected. A detailed explanation of the turn-on/off cirteria is discussed as well as corrections necessary for the X-position, bright count, and width count done on the second interval board.

A block diagram of interval board #1 is shown in Figure 1. This diagram shows the major circuit functions and critical signals. The major changes have dealt with the output state logic and, to some extent, the derivation of the turn-on/off signals OFFH and ONH. The first problem dealt with the turn-on signals. (See Figure 2 for a summary of the original on/off signals as reported in the last quarterly report.) Note that the right to left turn-on signal RLON is defined as:

$$RLON = \left(\sum_{p=0}^{L-1} B_{i,\,j-p} \geq K\right) \quad B_{i,j-p} = H_{i,\,j-p} \text{ or } C_{i,\,j-p}$$

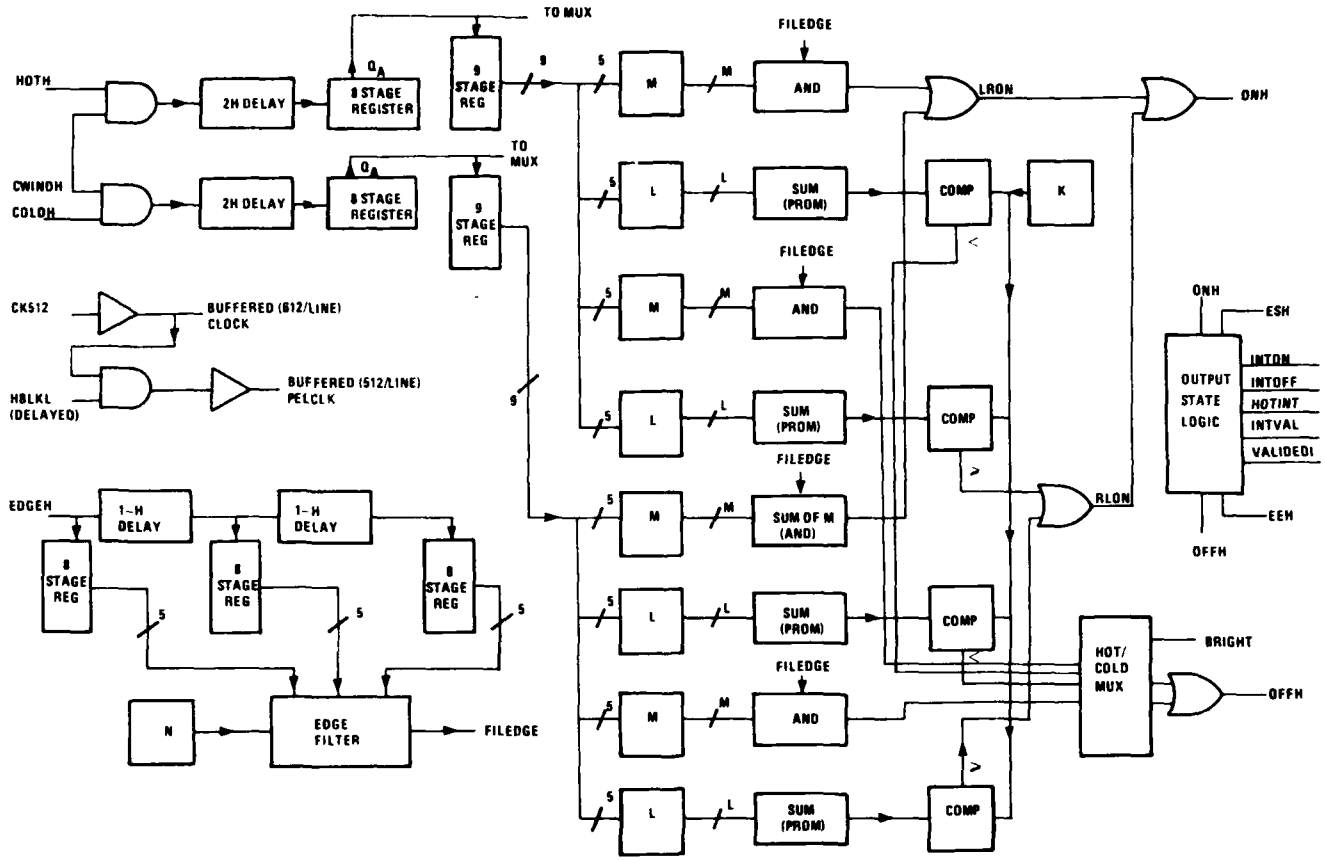


Figure 1. Interval Board #1 Block Diagram

- TURN ON AN INTERVAL IF (INTON)

$$\sum_{p=0}^{L-1} H_{i,j-p} \geq K \quad \text{(HRLON)}$$

OR

$$\sum_{p=0}^{L-1} C_{i,j-p} \geq K \quad \text{(CRLON)}$$

OR

$$\sum_{p=0}^{M-1} H_{i,j+p} = M \quad \text{AND FILEDGE TRUE} \quad \text{(HLRON)}$$

OR

$$\sum_{p=0}^{M-1} C_{i,j+p} = M \quad \text{AND FILEDGE TRUE} \quad \text{(CLRON)}$$

"(NAME)" IS LOGICAL SIGNAL NAME FOR TRUE RESULT

- TURN INTERVAL OFF IF (INTOFF)

$$\sum_{p=0}^{L-1} H_{i,j+p} < K \quad \text{(HLROFF)}$$

OR

$$\sum_{p=0}^{L-1} C_{i,j+p} < K \quad \text{(CLROFF)}$$

OR

$$\sum_{p=0}^{L-1} H_{i,j-p} < K \quad \text{(HRL1OFF)} \quad \text{(DELETED)}$$

OR

$$\sum_{p=0}^{L-1} C_{i,j-p} < K \quad \text{(CRL1OFF)} \quad \text{(DELETED)}$$

OR

$$\sum_{p=0}^{M-1} H_{i,j-p} = M \quad \text{AND } H_{i,j+1} = 0 \quad \text{AND FILEDGE (CHRL2OFF)}$$

OR

$$\sum_{p=0}^{M-1} C_{i,j-p} = M \quad \text{AND } C_{i,j+1} = 0 \quad \text{AND FILEDGE (CRL2OFF)}$$

Figure 2. Undirectional Interval Extraction

IN SUMMARY:

- **TURN ON IF**

  HRLON or CRLON or HLRON or CLRON

- **TURN OFF IF**

  HLROFF or CLROFF or HRL1OFF or CRL1OFF or HRT2OFF or CRL2OFF

FILEDGE (used to indicate occurrence of edge at start or end) is true if a or b is true below.

a. $\sum_{p=-2}^{2} E_{i-1,\,j+p} \geq N$ OR

$\sum_{p=-2}^{2} E_{i,\,j+p} \geq N$ OR

$\sum_{p=-2}^{2} E_{i+1,\,j+p} \geq N$

b. $\sum_{p=-1}^{1} E_{i-1,\,j+p} > 0$ AND

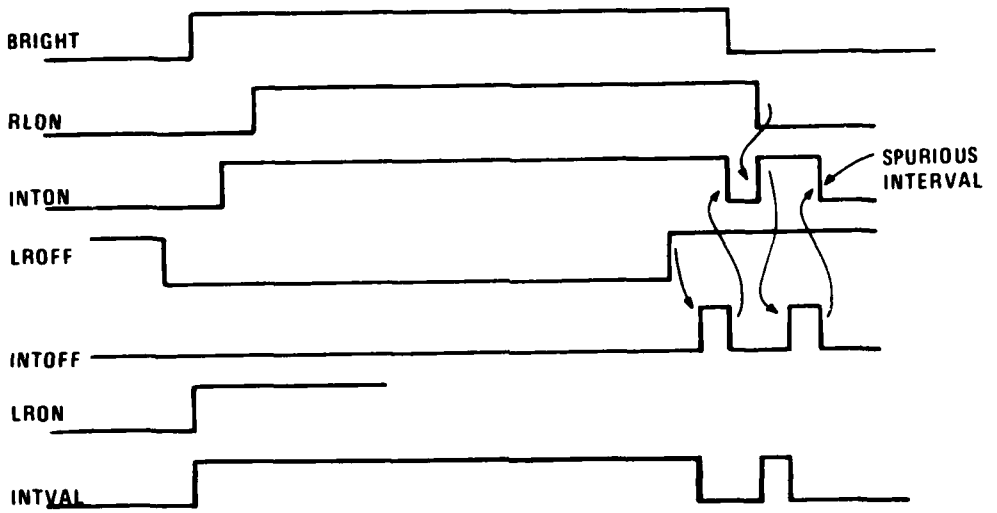
$\sum_{p=-1}^{1} E_{i,\,j+p} > 0$ AND

$\sum_{p=-1}^{1} E_{i+1,\,j+p} > 0$

Figure 2. Undirectional Interval Extraction (concluded)

This states that we turn on an interval if there are K or more brights over the last L pixels. The test results from logical inversion of the right to left turn-off test as used in the simulation. This signal will properly turn on an interval which has only an edge at the end; however, it will also confuse the state logic since it can also call for interval turn-on after a valid interval turn-off, as shown in Figure 3. This situation occurs when we apply the RLON test on an interval without an edge-at-end. The effect is to generate a spurious two-pixel interval, which is only an annoyance, since it would not usually be validated as long as FILEDGE were not true during the spurious interval time. The whole problem is caused by the fact that RLON overlaps the true interval end by L-K pixels. To alleviate this, the L and K parameters in the HRLON and CRLON circuits are forced to be equal, hence a K - K = 0 pixel overlap. Since the implementation uses the same K in all tests, we split the L blocks into LON and LOFF, where LON = K is required. This modification clears up the spurious interval problem but also makes the actual turn-on for EE intervals more stringent. (K of K brights instead of K of L.) It also makes the PROM implementation of a K of K test somewhat inefficient. (AND gates would suffice.)

The second problem encountered was another spurious interval phenomenon, specifically a spurious interval dropout. The cause was found to be the RL1OFF signals in Figure 2. The RL1OFF signal was incorrectly included as a turn-off criteria and had no basis in the simulation. Removing the signal from the turn-off OR gate solved the problem.



K=3, L=4, M=2

Figure 3. Example of Problem with Using RLON Without Modification (Assume ES only)

The next problem discovered was one relating to the ESH and EEH signal timings. These signals were being reset one clock cycle after INTOFF, which led to excitation conflicts. Furthermore, the VALIDEOI signal was being lost because of improper timing of INTOFF, ESH, and EEH. To correct the problem, EEH and ESH are now reset after INTON returns low, and a delayed INTOFF signal has been developed to sync-up with EEH and ESH. The loading of the first level feature latches is now accomplished on the falling edge of FIFOLOADH, which is also the edge at which ESH and EEH are reset. (This satisfies the LS374 set-up time requirements.) Of particular importance is the fact that a VALIDEOI pulse will not be generated until two pixels after an interval end.

The last problem to be addressed here is in regards to the counter offsets introduced by processing lags and the turn-on/off decision criteria. Analysis of the revised turn-on/off criteria shown in Figure 4 will give the required correction terms for the C-position, bright, and width counters. This correction has been made with the result shown in Figure 5. We see that the X position offset is strictly related to the ES bit. However, detailed analysis shows that the width and bright counts must be dithered by a term depending upon both EE and ES bits. A full correction scheme might look like the one in Figure 6. This scheme requires four adder chips, three quad 2-to-1 muxs, and two 8-pole DIP switches. The correction would be added to the stable count just prior to latch loading. Maximum error with the EE correction excluded is K-1 pixels.

$$RLON = \left[\bigwedge_{P=0}^{K-1} B_{i,j-p} \geq K \right]$$

-K OR MORE CONSECUTIVE BRIGHTS

$$LRON = \left[\bigwedge_{P=0}^{M-1} B_{i,j+p} = M.\ AND.\ FILEDGE \right]$$

-M BRIGHTS WITH EDGE

$$LROFF = \left[\bigwedge_{P=0}^{L-1} B_{i,j+p} < K \right]$$

-LESS THAN K BRIGHTS IN L PIXELS

$$RLOFF = \left[\bigwedge_{P=0}^{M-1} B_{i,j-p} = M.\ AND.\ FILEDGE.\ AND.\ B_{i,j+p} = 0 \right]$$

-M BRIGHTS WITH EDGE AND NEXT PIXEL OFF

THESE TESTS PERFORMED FOR $B_{i,j} = H_{i,j}$
AND FOR $B_{i,j} = C_{i,j}$

Figure 4. Revised Turn-On/Turn-Off Tests

SUBTRACT FROM X-POSITION:

ES=

0	$L_{ON} + D_1$
1	$M + D_1$

D_1 = PROCESSING LAG IN INTERVAL BOARD
= 4 PIXELS AT THIS TIME

ADD TO WIDTH COUNT, BRIGHT COUNT:

EE \ ES	0	1
0	DON'T CARE	M + K-5
1	L_{OFF} 2	M-2

CURRENTLY

$L_{ON} = K = 3$

$N = 3$

$M = 2$

$L_{OFF} = 4$

Figure 5. Counter Correction Tables

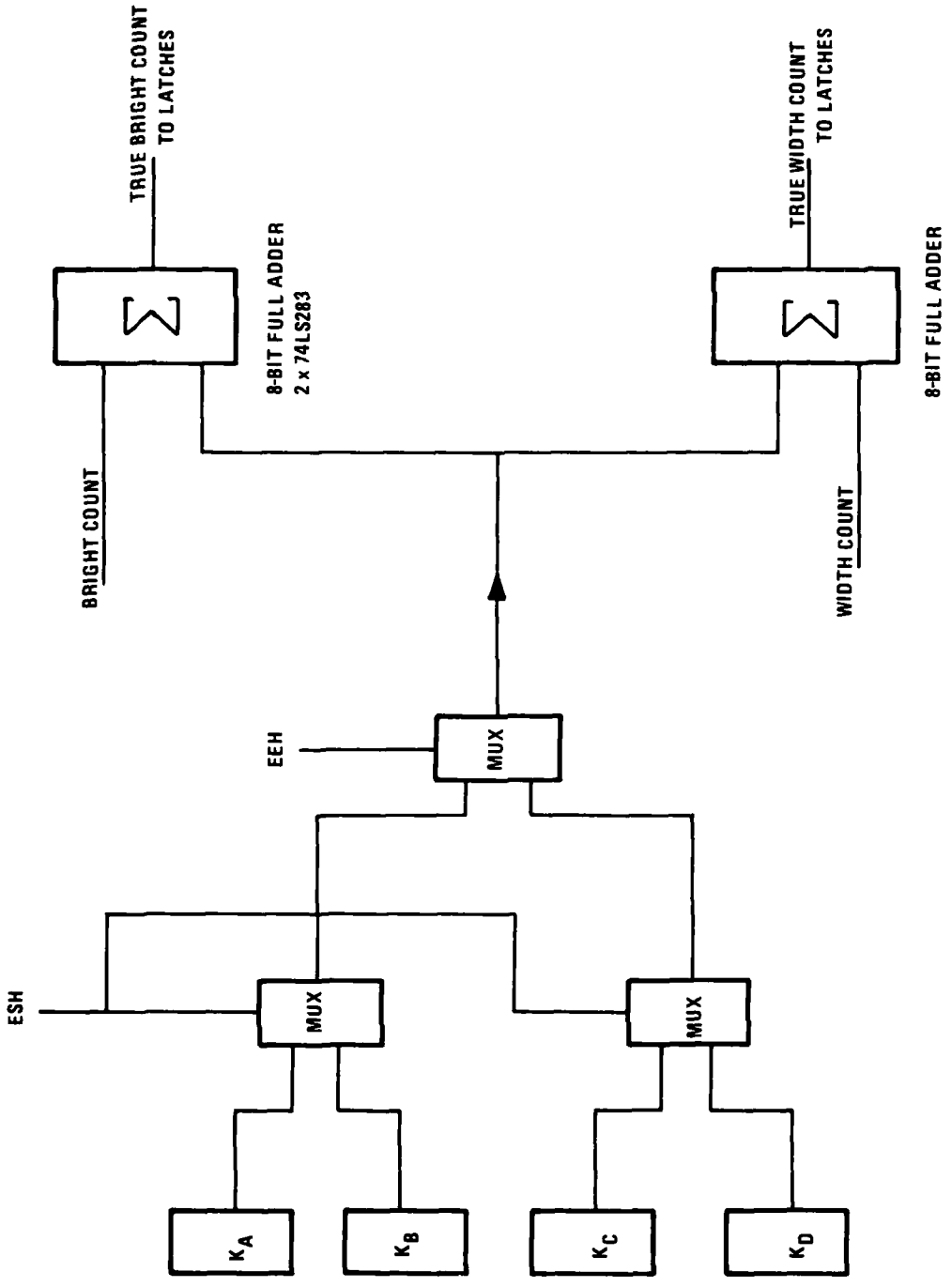


Figure 6. Revised Width, Bright Count Correction

GLOBAL GAIN/BIAS CONTROL CIRCUITRY

The global gain/bias control circuitry now includes a Synertek 6503 Processor Chip instead of a Intel 8748. This approach was chosen primarily because of the availability of test equipment which supports the 6503. A connector at the top of the global gain/bias board is used to connect a SYM-1 microcomputer for software development. Following is a description of the system and the algorithm used for complementing global gain/bias.

System Hardware

The global gain/bias system has the function of controlling the gain and bias of an infrared camera (FLIR). The entire system consists of: an input circuit, which counts the number of exceedances of a set threshold in the input signal from the FLIR; an output circuit, which outputs the control signals to the FLIR; and a control circuit, which interfaces to the input and output circuits and executes the control algorithm.

Input Circuit--The input signals from the FLIR consist of video in (VIDBL), video clock (CK 455H), and a frame sync (VDRVH) signal. The input video signal is compared with an upper threshold and a lower threshold by two comparators. The output of each comparator is clocked into a counter by the video clock. These counters then contain the number of upper and lower threshold exceedances. Two optional input signals are also available. These two signals are multiplexed with the high order "carry" bits of the two

threshold counters. This allows the control circuit to access all four signals: "carry" bits when S = 0 and optional signals 1 and 2 when S = 1. The multiplexer also provides a clear signal for the counters when S = 1. The input circuit is shown in Figure 7.

Output Circuit--The output circuit consists of two digital to analog converters with a I/V converter-buffer amplifier for each. These two signals connect to the gain and bias controls of the FLIR. The output circuit is shown in Figure 8.

Control Circuitry--The control circuitry consists of: a SY-6503 microprocessor, which is the master controller; a SY-6522 VIA, which interfaces the processor with the input circuit; a SY-6532 RAM, I/O, timer, which provides scratch pad memory for the processor and interfaces it to the output circuit; and a 2708 EPROM, which contains the control program. The diagram for the control circuit is shown in Figure 9. The devices are decoded according to the memory map shown in Figure 10. This allows for zero page RAM and stack area in the SY-6532; it also provides for vectors at the top of memory to be put into the EPROM.

The system also has an interface which allows it to be connected with a SYM-1 single board computer for use in software development. This allows the SYM-1 to access the locations 000-FFF in the control circuit by addressing locations 1000-IFFF. It should be noted that control circuit devices will decode any address X000-XFFF even though the data bus might not be enabled. For this reason, it is advisable to place all control programs in locations 0C00-OFFF when running them with the SYM-1, with the vectors set accordingly. Also, when running with the SYM-1, the SY-6503 must be removed, and the clock (Ø 2) connected to external source (on board jumper).

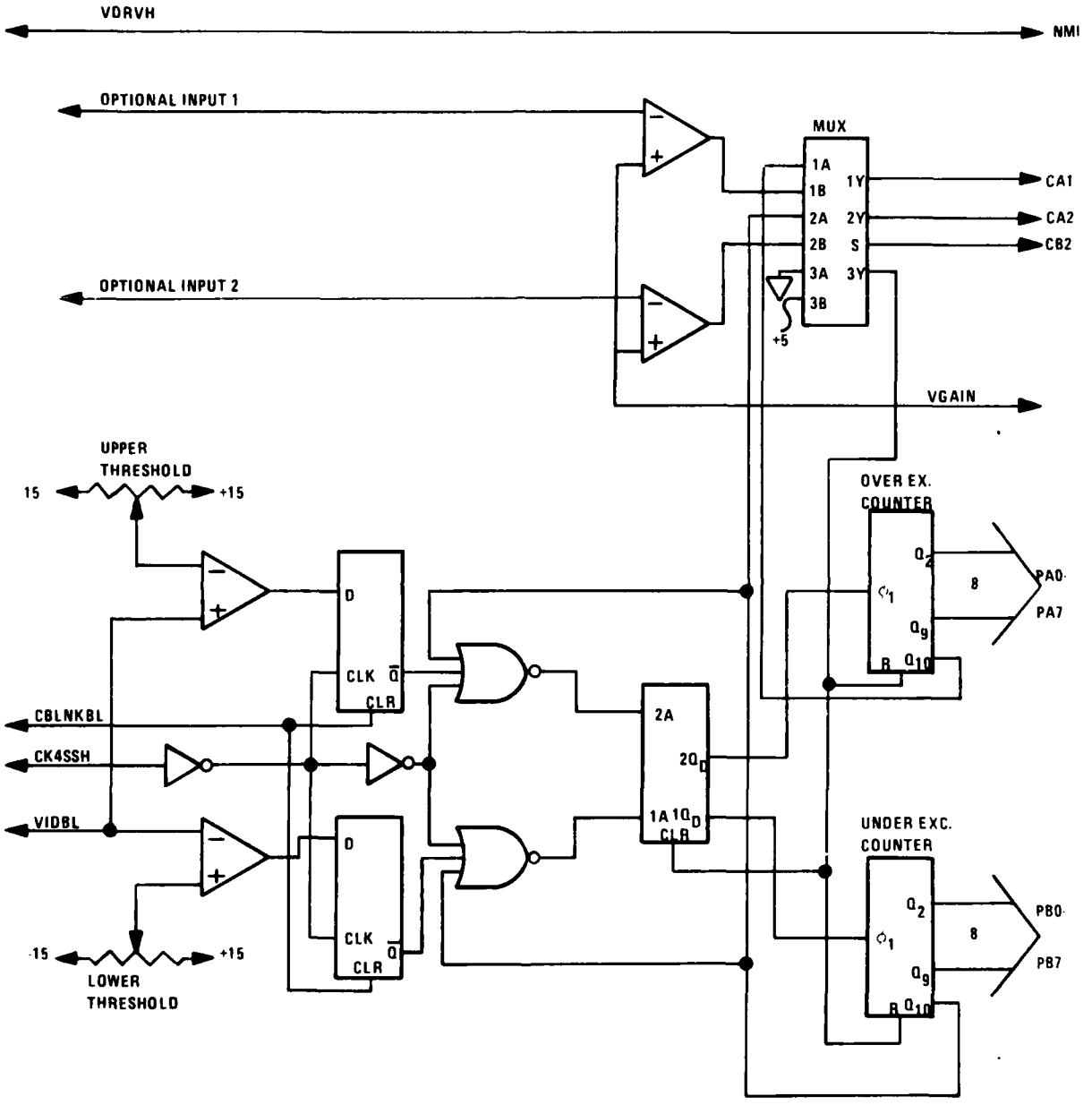


Figure 7. Input Circuit

SY-6532

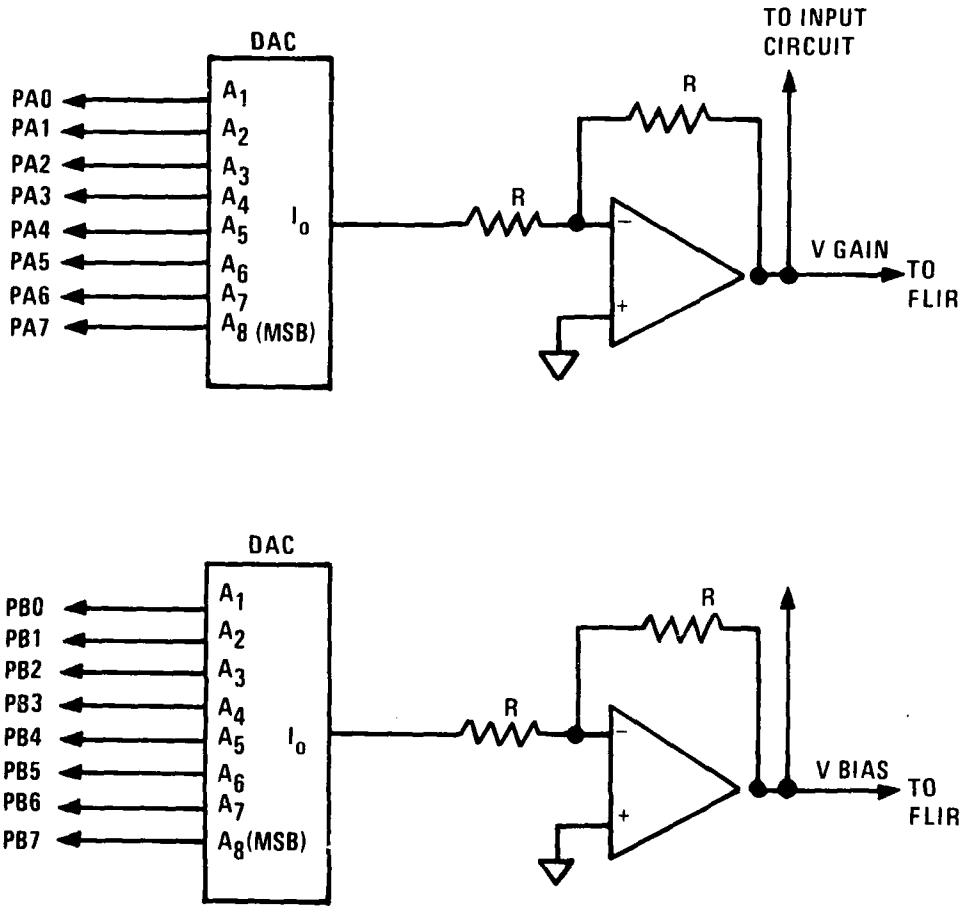


Figure 8. Output Circuit

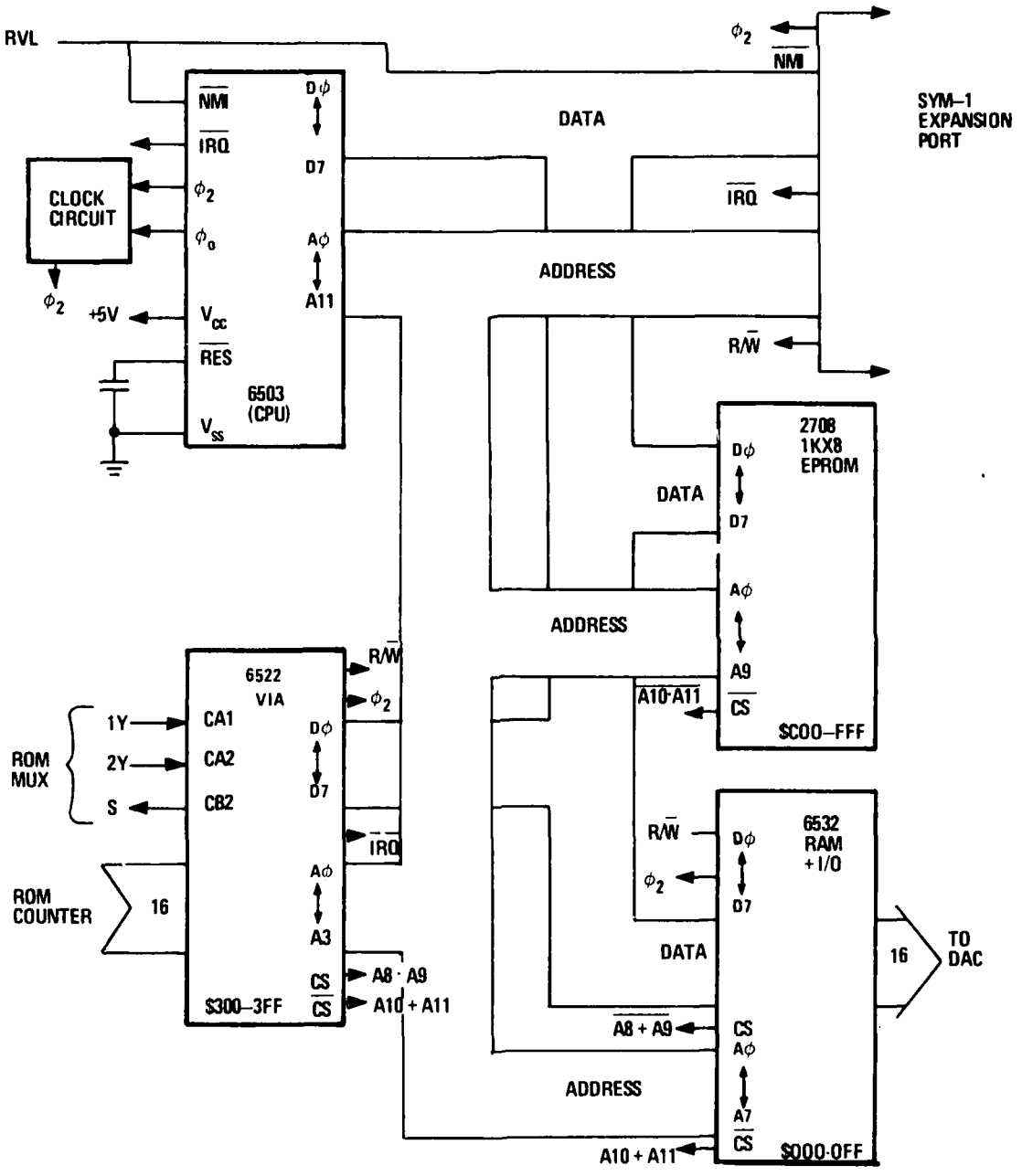


Figure 9. Global Gain/Bias Control Circuit

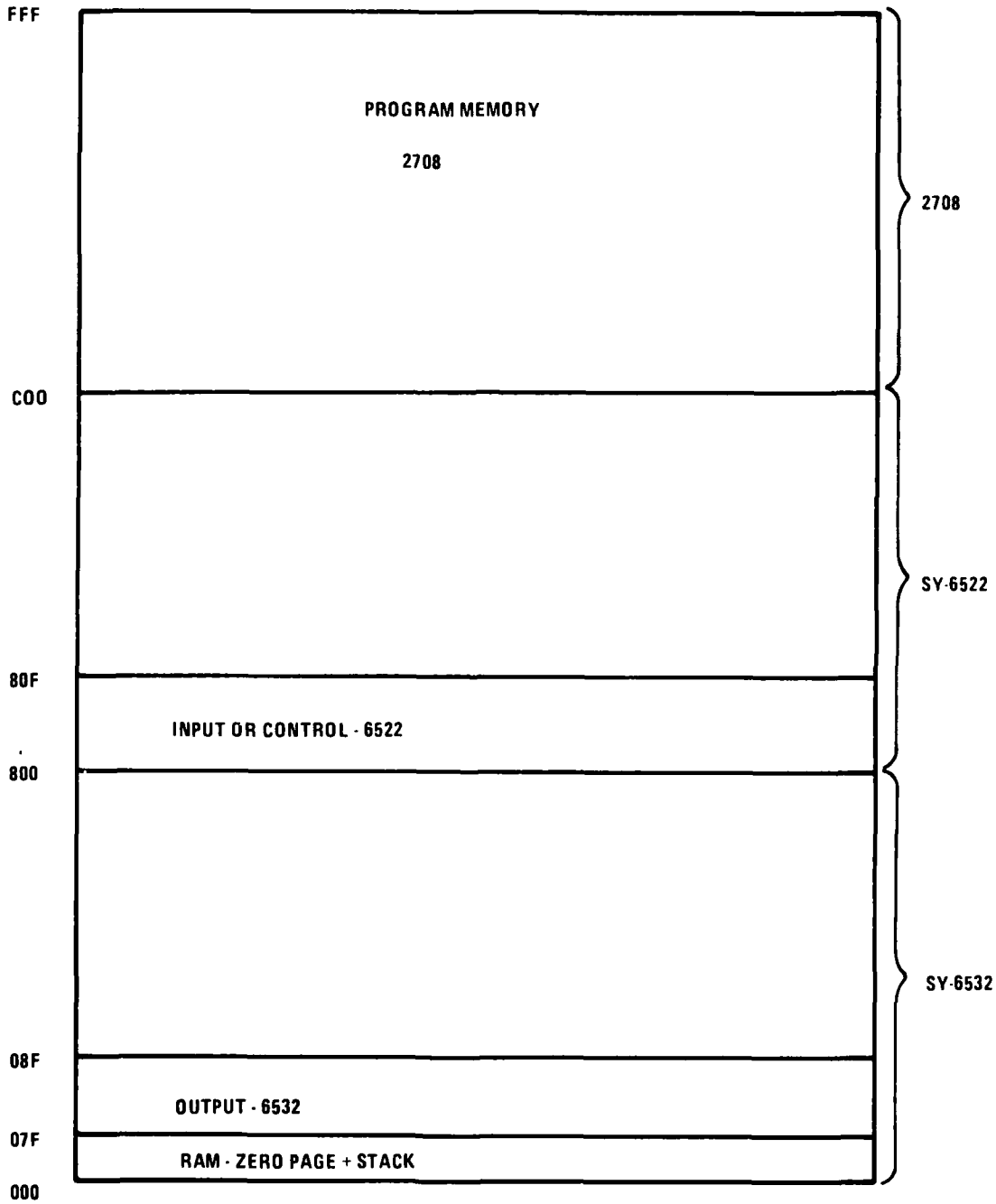


Figure 10. System Memory Map

Software

The basic control is implemented by a software algorithm that runs in the SY-6503 CPU. The algorithm makes control decisions based on the number of upper and lower threshold exceedances per video frame.

On restart, the processor initializes its status register and stack pointer, initializes the I/O registers and registers used by the program, and finally outputs maximum gain and nominal bias to the FLIR. The main program then waits for a WM frame settling time and a IM frame averaging time. The fr e sync signal (VDRVL) is used as an NMI interrupt. During the NMI routine, the frame is counted and the count is compared with the specified wait time (WM). If the wait time condition has been met, the NMI adds the under-and over-exceedances counts of the next (IM) frames.

The VIA is configured so that an overflow of the threshold counters will cause an IRQ interrupt. The IRQ routine which will add the overflow to the sum for that frame is then written.

After (NM) frames have been counted and the exceedance counts of (IM) frames added together, the sums are divided by IM (shifted $\log_2$ (IM) times right) to obtain an average exceedance count per frame.

After the data have been collected, the program executes the control algorithm whose state diagram is shown in Figure 11. The algorithm is implemented using the flow chart of Figure 12.

Figure 13 also shows the way program memory is allocated.

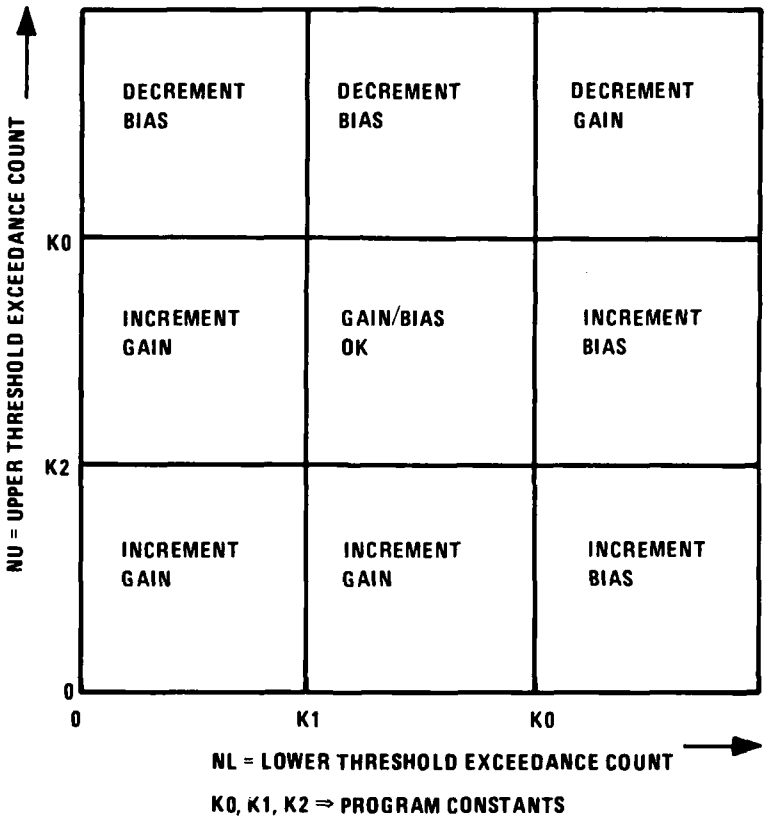


Figure 11. Control State Diagram

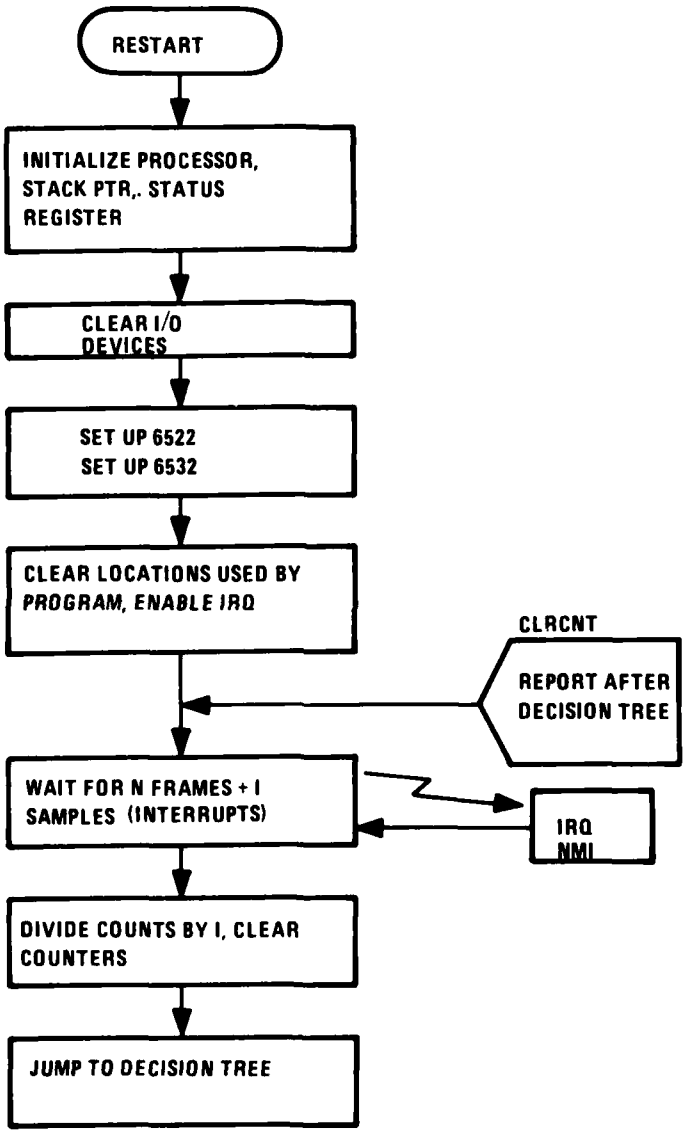


Figure 12a. Restart Routine

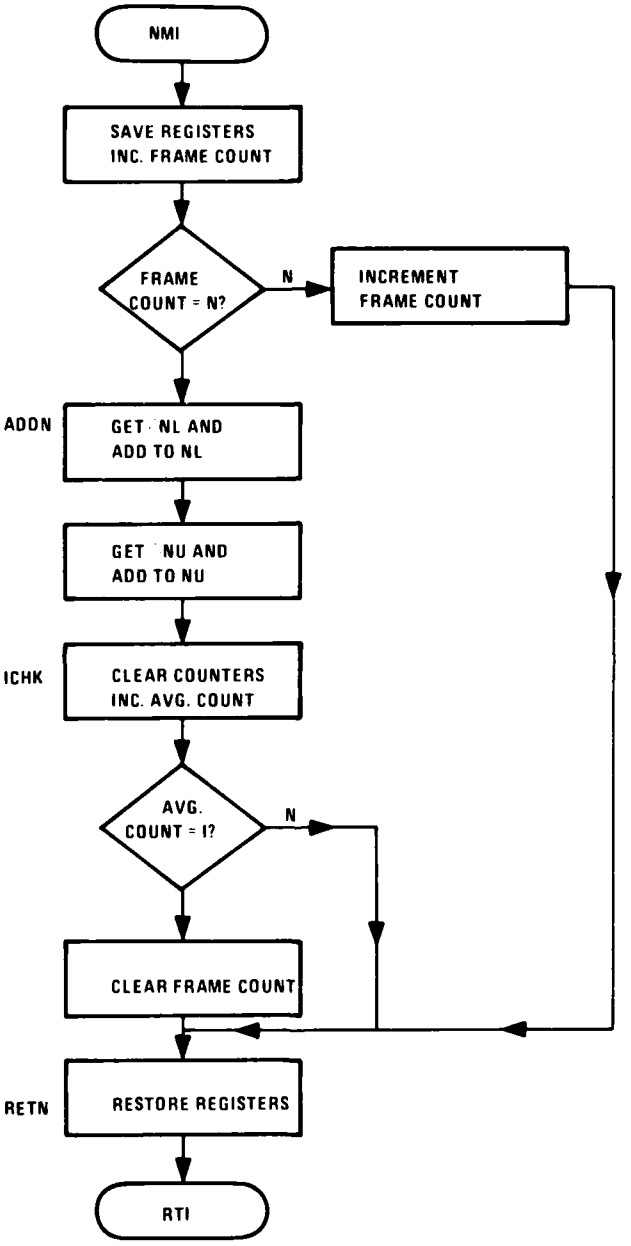


Figure 12b. NMI Routine

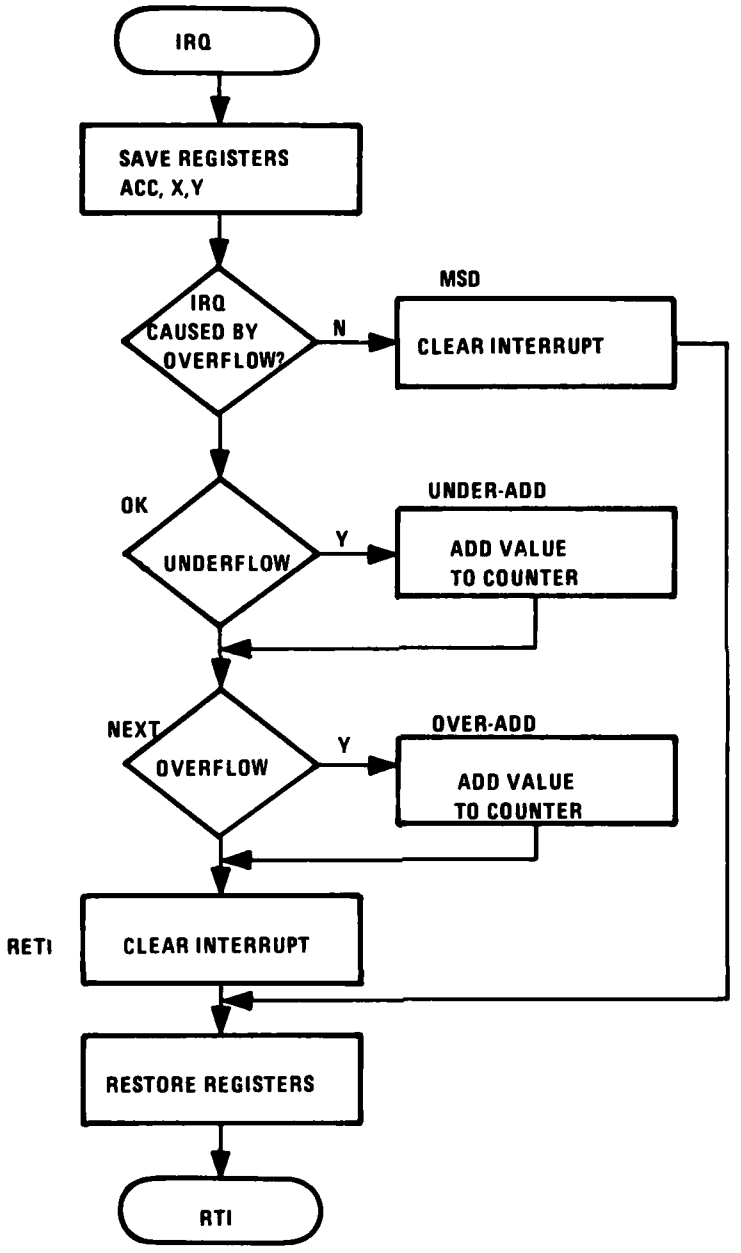


Figure 12c. IRQ Routine

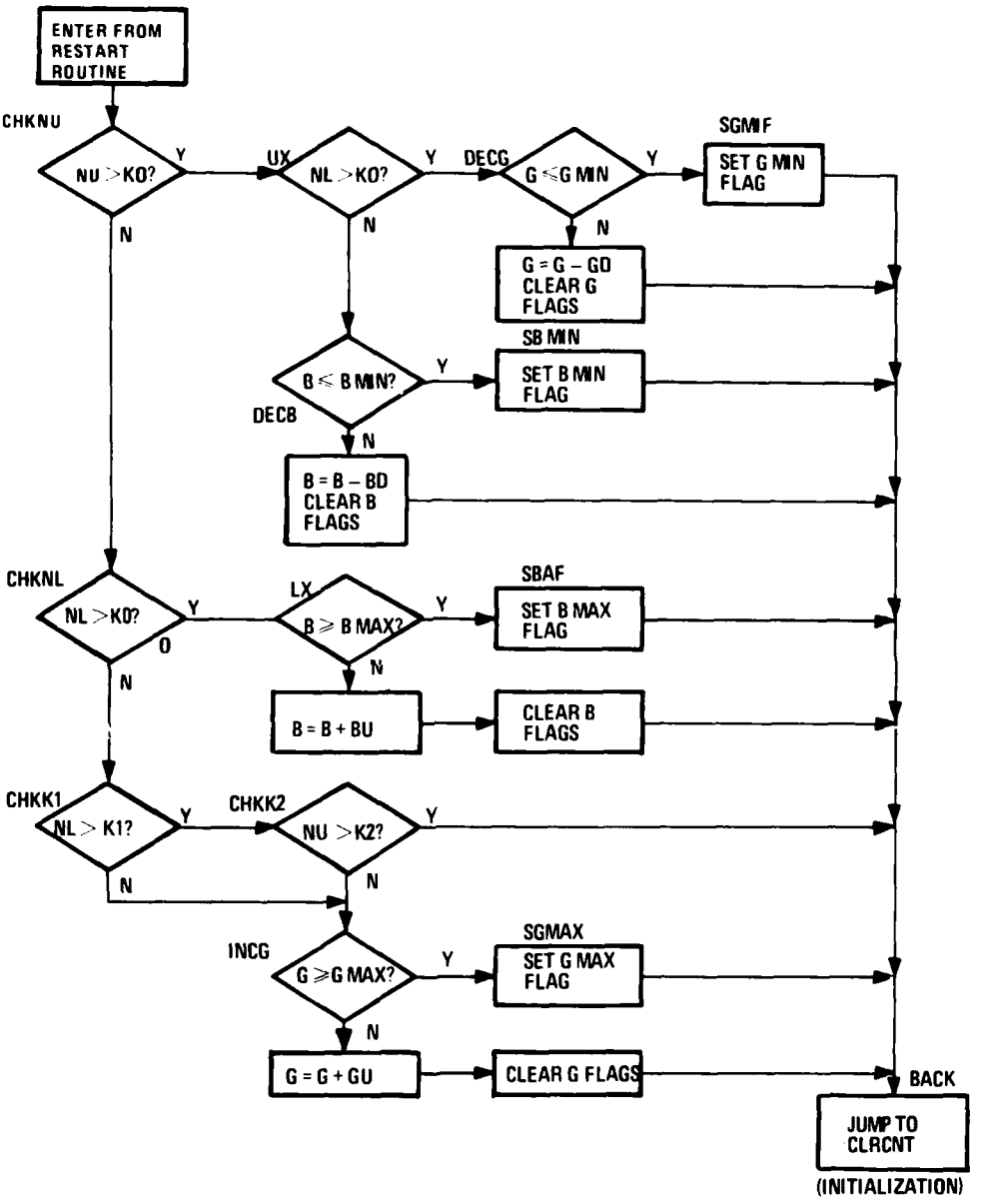
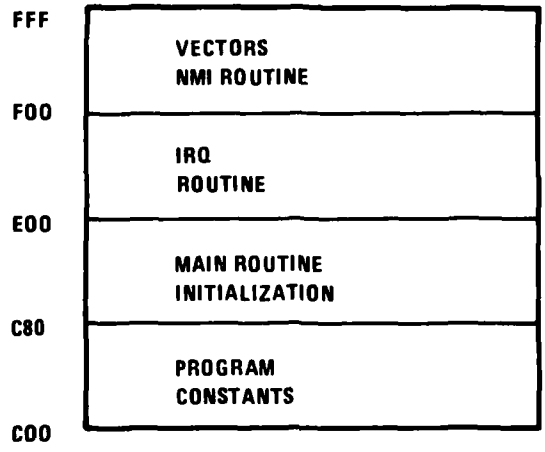


Figure 12d. Decision Tree



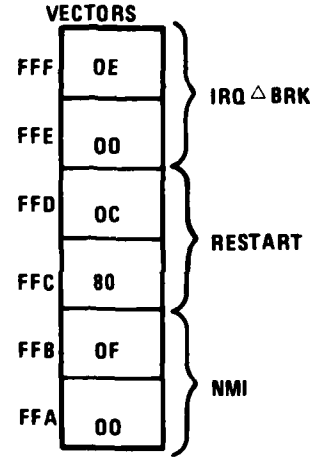


Figure 13. Program Memory Assignment

FLYABLE HARDWARE MODS

Three changes were made to the PATS system which enable the entire system to be flyable. The primary concern is that only 400 Hz power is available in the helicopter. Hence CPU2 (a DEC LSI 11/2) must be eliminated from the PATS flyable hardware since it requires 60 Hz. The software functions in CPU2 that must be transferred are the inter-frame analysis and symbol generation. The additional hardware that is needed includes a CPU1 to symbol interface, a laser range finder interface, and a remote test box. Each of these three items is discussed below.

CPU1 to Symbol Interface

Conceptually the CPU1 to symbol interface is the same as the CPU2 to symbol interface. The primary difference is that the bus interface to CPU1 requires different integrated circuits. Instead of bus transceivers, bus protocol chips, and interrupt chips, the interface is now implemented with decoders and buffers to the data and address bus.

The device address for the symbol generator relative to CPU1 address space will be:

SEL0L	7000	X Address
SEL2L	7001	Y Address
SEL4L	7008	Vector Address Read/Write
SEL6L	7009	Graphics Control

It should be noted that the operation is the same as that of CPU2, but different addresses are involved. The reader is referred to the 6th PATS quarterly report (6 April 1979) for explanation of how the symbol generation and vector generation are done. Each specific select (SEL) operation and the command formats are the same as those discussed in the earlier report except as noted below. Table 2 is the vector read/write (SEL41) command format and Table 3 is read vector data command format.

Laser Interface

The laser range finder interface associated with the LOHTADS is quite simple. The data comes in a serial format and consists only of range data. It is boresighted to the center of the field of view of the LOHTADS FLIR. When range data is sent to CPU1, only one 16-bit word is required. Figure 14 shows the range finder interface.

Data and clock are received in parallel. When data are valid, the read enable line goes high, allowing the data to be clocked into the serial to parallel converter. At the end of the data conversion, the read enable goes low, triggering a one-shot pulse which transfers the data to the latch where they are held for CPU1 to read at the beginning of a TV frame. Not shown is some logic that stops CPU1 from reading the data when the transfer is taking place.

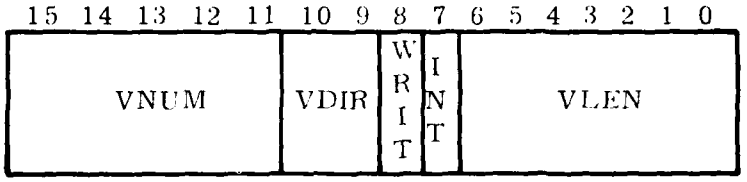
The use of this range data is discussed in the software section of this report.

TABLE 2. VECTOR READ/WRITE COMMAND FORMAT (SEL4)

SEL4

Output

Data To Interface



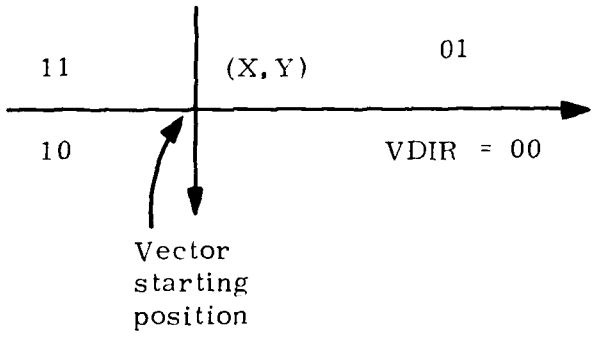
INT = 1 Intensify vector if WRIT = 1

= 0 Erase vector if WRIT = 1
If WRIT = 0, then INT causes no-operation

WRIT = 1 Write vector

= 0 Read vector

VDIR = Vector direction from starting point
Bit 9 = 0-left to right
Bit 9 = 1-right to left
Bit 10 = 0-Top to bottom
Bit 10 = 1-Bottom to top



VNUM - Vector number from 0-31 which selects one of the preprogrammed vectors

VLEN - The length of the vector in pixels, up to 128 elements

TABLE 3. READ VECTOR DATA COMMAND FORMAT (SEL4)

SEL4

Read Vector Data

Input

Data To Interface

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DONE														DATI	VDRV

DATI--Value of graphics memory at the last point in a vector that was read.

DONE--A "1" when any read/write operation to SELØ, SEL2, SEL4 has been completed (i.e., operation done signal). A "0" means not done. The "DONE" signal is checked after vector read/writes. It is always a "1" after X and Y address writes. Doing a SEL4 read does not effect "DONE".

VDRV--A "1" during the vertical drive pulse of the video. A "0" at other times. This allows field synchronization.

Test Box

For flight test purposes, it is necessary to have a remote test box with which the pilot can control some of the operation of the PATS hardware. Also there are conditions under which the laser range finder cannot be fired because of safety restrictions. Hence, it is desired to provide both a manual input and a transponder interface whose output looks like the laser range finder. In this section we address the relative connections to the test box, the test box layout, and the interfaces.

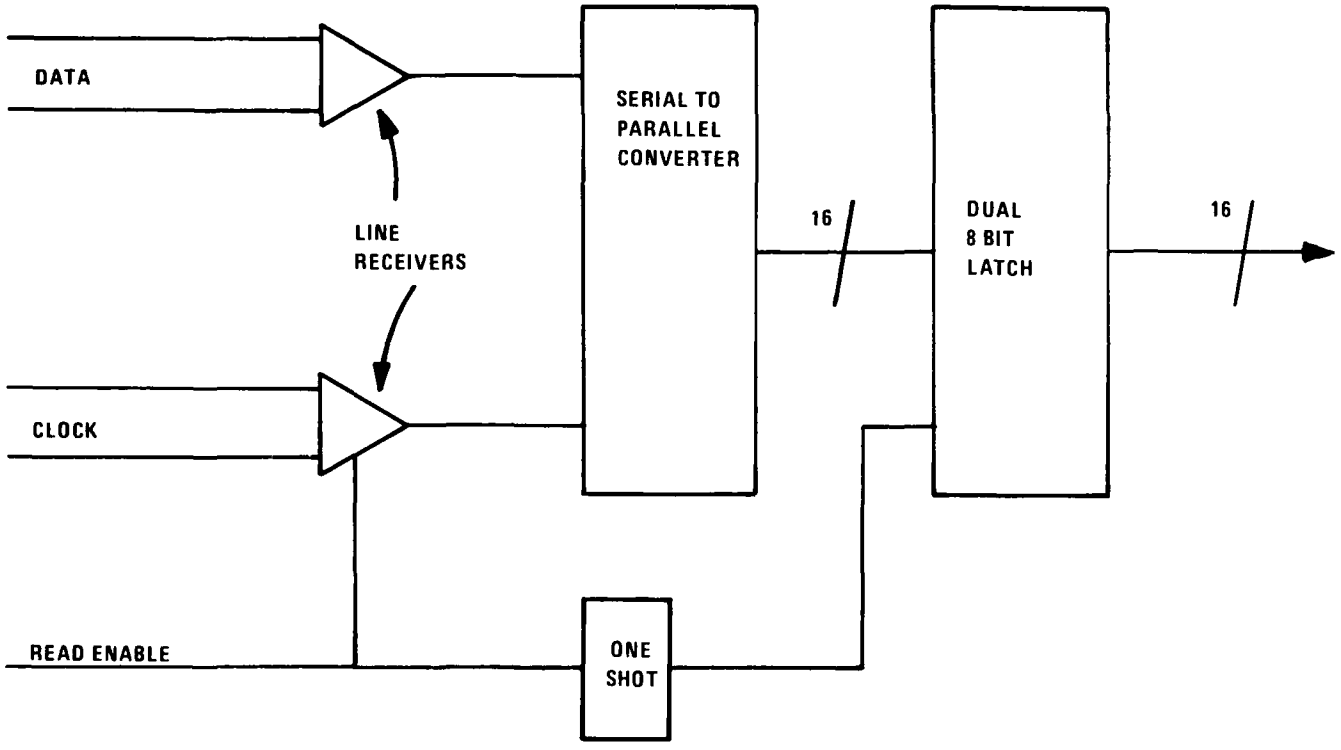


Figure 14. Laser Range Finder to PATS Interface

Figure 15 shows the interconnections between the PATS hardware, FLIR, and transponder range. The connector PATS-J1 to Test Box-J1 contains the logic signals to control CPU1 operation as well as the gain/bias voltage and video switching controls. PATS-J2 to Test Box-J2 is the range signal only. This cable can be removed and the laser range finder can be connected to PATS-J2. Also there are several reserved pins on PATS-J2 for CPU1 checkout. Test Box-J3 goes to the transponder receiver output. Only four BCD digits are assumed to be coming from the transponder. Test Box-J4 routes the control signals to the FLIR.

Figure 16 shows the test box layout. Manual switches provide for range input under the condition of voice only recording of range. Video switches are also provided to gate the video going to the target screener and to the monitor. The other switches are primarily for CPU1 control and will be in a fixed position during flight tests. The reset button allows for manual interruption of PATS during the flight tests. The LED indicates that power is on or off. Initial power-on is located on the PATS hardware, however.

The test box contains one electrical card which takes the BCD input from the switches or the transponder output and converts the BCD input to binary. Figure 17 show a block diagram of the functions provided on the card. When the data are converted, they can be sent to the PATS main chassis by being converted to serial format and sent over at the appropriate rate. Line drivers are used to send the signals over cables.

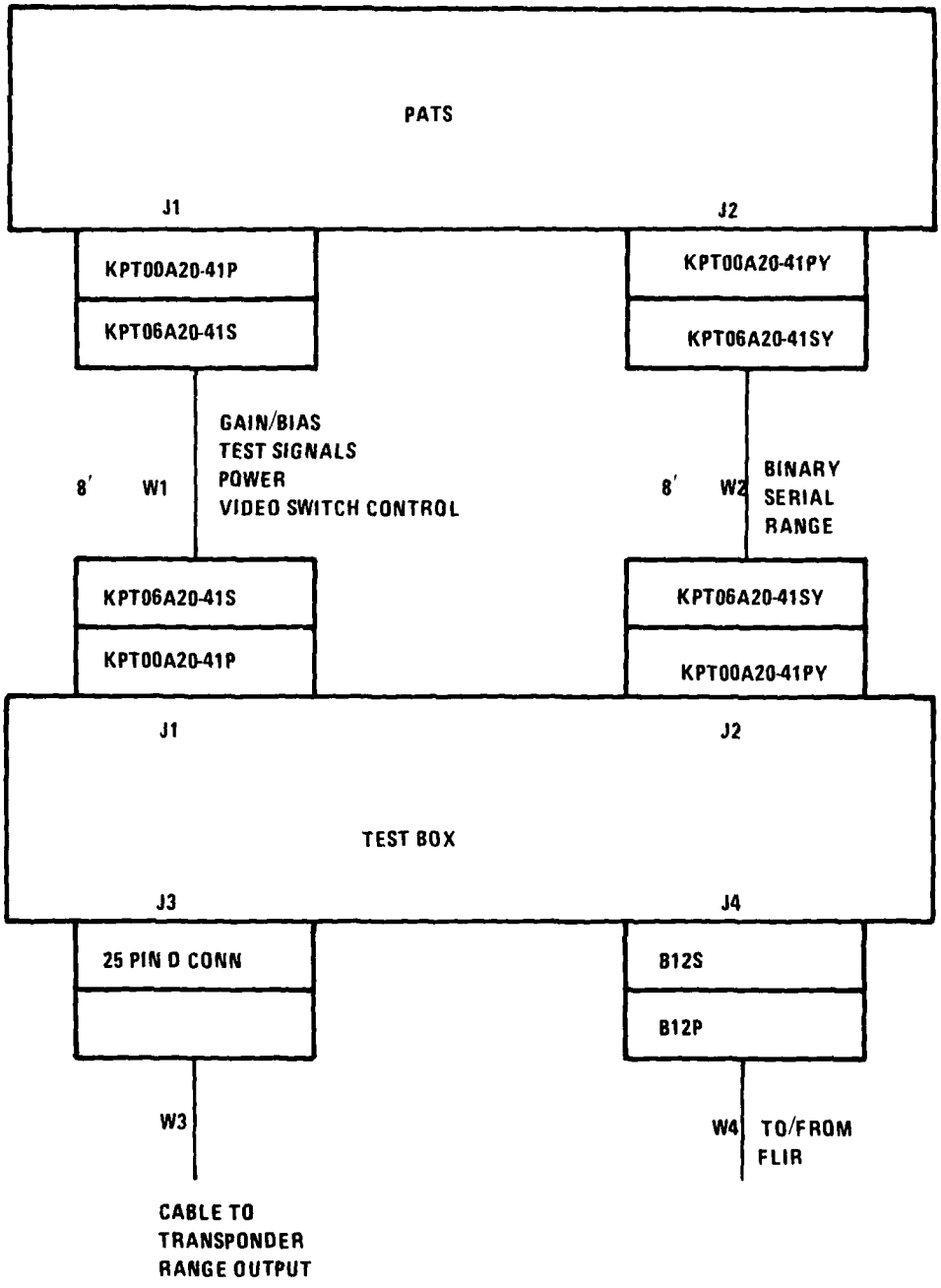


Figure 15. Layout or PATS & Test Box Cabling Diagram

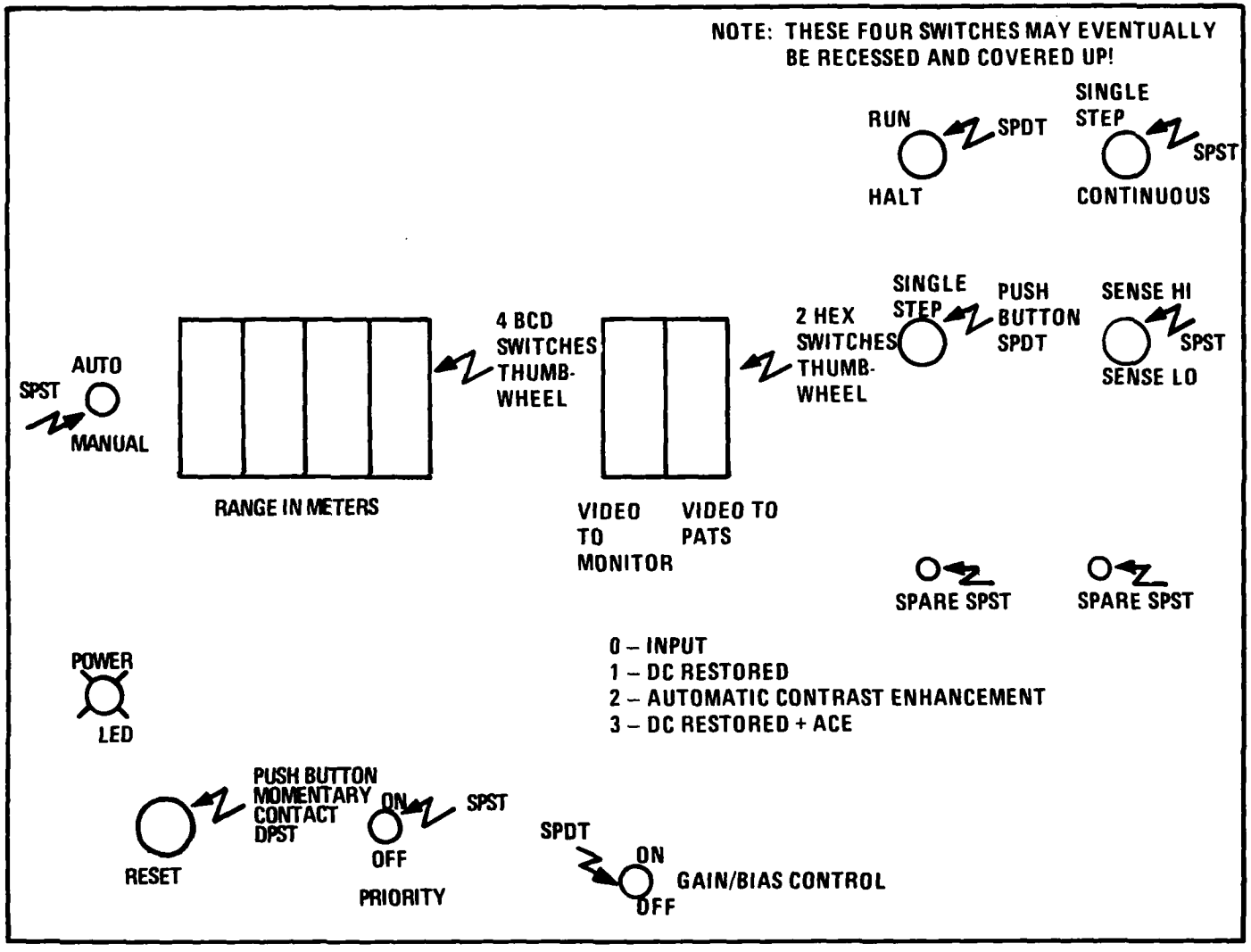


Figure 16. Test Box Layout

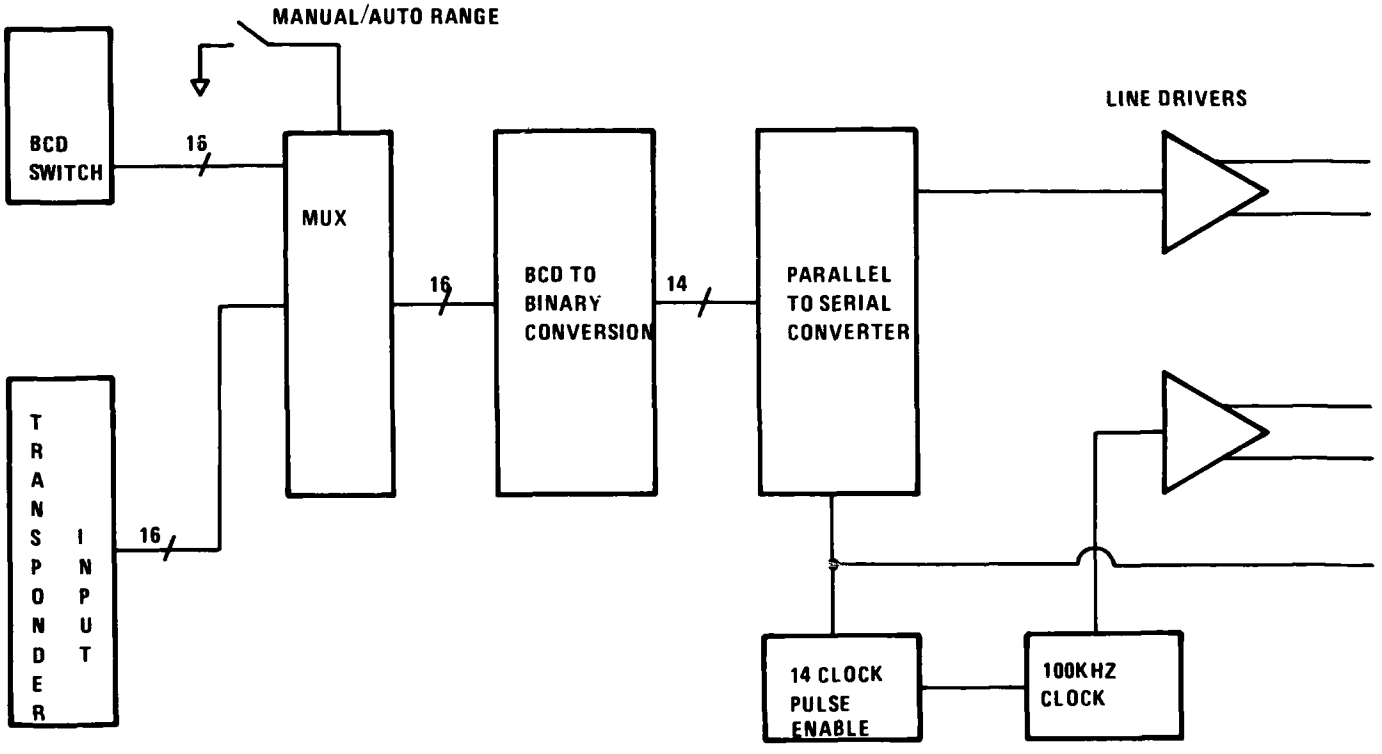


Figure 17. Test Box Transponder to Serial Converter

SECTION 3

SOFTWARE STATUS

In this section we describe how we plan to use range data initially and some documentation which has been developed for CPU1 diagnostics. Abstracts for the rest of CPU2 software are also provided. Some of the CPU2 software will not be used since much of it now is being written in microcode in CPU1. However, this software can be used during training and checkout. The section begins with a statement on the microcode status.

MICROCODE STATUS

During this reporting period, the microcode has been in the process of debug. In general, no new bits have been added to the microword format. We have concentrated on the microcode to include range and interframe analysis. Data had to be hand derived initially, but this only affects the checkout of the PATS hardware and much of the software will not be used except in the initial checkout.

Many problems have been uncovered in our initial checkout including how to talk to specific devices, how CPU1 is interrupted at the end of each scan line, and how to get a near frame and memory addressing for reading and writing. Based upon the addition of new software, the definition file has changed. It is not presented at this time since it is still being changed to accomodate the problems encountered.

RANGE SOFTWARE

The PATS system must be capable of using range as an aid in identifying targets. We have already described hardware which will accept range in a binary serial format. However, a problem occurs if no range information exists since the data latches may come up in an unknown state. Hence, the test box will have to be connected at all times such that a value can be put in manually. All zeros will indicate no range data are present.

As part of the training procedure, we will input range and aspect angle once per frame. Note that this is relative to the training data we have received and not the general case. This should help speed the training process. For training purposes, the range will be entered manually from the DEC keyboard and not from the test box. The range is on the audio channel as a voice calling out ground range; whereas, the aspect angle is obtained from the flight data sheets.

The aspect angles nomenclature will be as follows:

0 - Unknown
1 - Right side view
2 - 3/4 right front
3 - Front
4 - 3/4 left front
5 - Left side
6 - 3/4 left rear
7 - Rear
8 - 3/4 right rear

This is the order in which the flights for the test data were flown.

The range data will affect the clutter classifier initially. At the present time we will not use range in the KNN classifier or the interframe analysis.

The approach we will use for PATS will be as follows: At the maximum range (8500 meters), we will determine the minimum target size of all the targets and all aspect angles. Similarly, we will determine the maximum size of all the targets at that range. All this information can be obtained from the training data. For every 500 meters we will determine the minimum and maximum target size areas over all the classes as derived by the PATS hardware. At this point we will have a table of data as follows:

Range	Min Target Size	Max Target Size
8500		
.		
.		
.		
1500		

A factor such as 0.85 times the minimum and 1.15 times the maximum will then be stored in 30 locations of memory in range order so that range does not have to be stored.

The algorithm will then use range to enter a table to determine the thresholds for the minimum and maximum size for that range. If the object size is less than the minimum or greater than the maximum, it should be rejected as clutter.

The nearest lesser range should be chosen before making the comparison. A general flow-chart is shown in Figure 18.

This approach uses a few memory locations in memory #1. Eventually this may expand into a table which consists of:

Range, Target 1, 8 aspects, Target 2, 8 aspects, etc.

For four targets, this would require about 500 words. The KNN classifier could then compare moments results and size to determine target classification, i.e., the decision would be: the size is like a tank at aspect a or a jeep at aspect b. Based upon other data, a decision could then be made.

At this time we will not implement this type of structure because of the amount of memory and programming involved.

DIAGNOSTICS FOR CPU1

This is the documentation for the CPU1 diagnostic software which runs under control of CPU2. The "Use Matrix" in Table 4 lists the main programs which can be invoked from the LSI-11. In Table 5 is a list of all programs and a sentence describing each. Since some programs need more explanation, they are discussed individually.

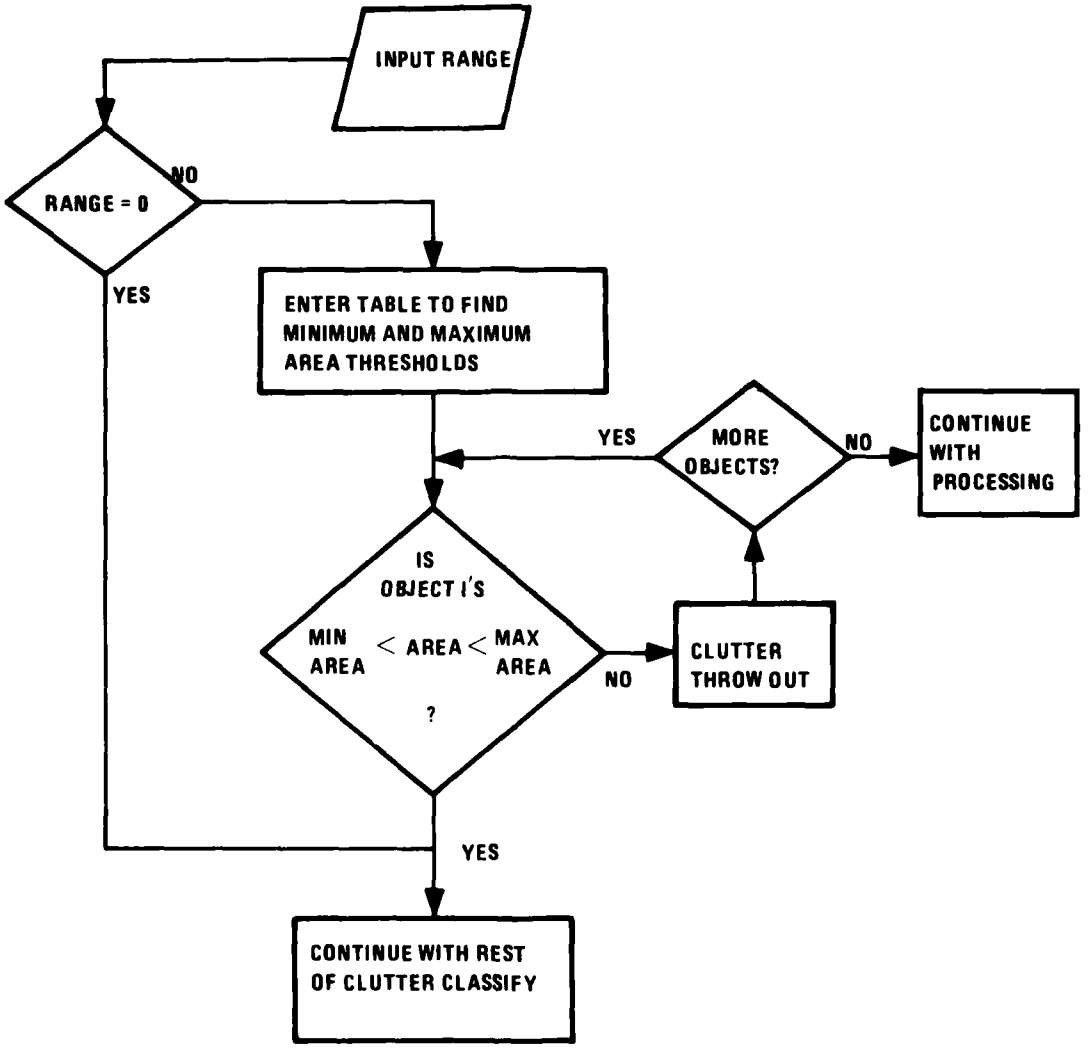


Figure 18. Generalized Flow Chart

TABLE 4. USE MATRIX

Sub-routines	Main Programs							
	DUMPM1	DUMPM2	D2M2DT	D2M1WI	FORBIN	FORMIN	MEMTE1	MEMTE2
DMAOUT	X	X	X	X			X	X
EREPRT							X	
EREPR2								X
ISRER	X	X	X	X			X	X
ISRI	X	X	X	X			X	X
MACLIB	X	X	X	X			X	X
MEMEX1							X	X
MEMEX2								X
PAIN	X	X	X	X			X	X
ORUM					X	X		
UPIT					X	X		

TABLE 5. DIAGNOSTICS FOR CPU1

PROGRAM	
DMAOUT	performs a DMA operation for sending to CPU1
DUMPM1	prints contents of user specified memory #1 locations or CPU1 registers
DUMPM2	prints contents of user specified memory #2 locations
D2M2DT	reads a file of data, writes it in memory #2
D2M1WI	reads a file of data, writes it in memory #1
EREPRT	reports defective memory #1 locations
EREPR2	reports defective memory #2 locations
FORBIN	formats a file of bin data
FORMLN	writes a file containing a vector of data keyed in
FORMIN	formats a file of interval data
ISRER	prints error messages for ISR1
ISR1	interrupt service routine for DMA
MACLIB	library of macros from file MACROS
MACROS	text file of macros used by several routines
MEMTE1	requests parameters for MEMEX1
MEMTE2	requests parameters for MEMEX2
MEMEX1	performs memory test on memory #1
MEMEX2	performs memory test on memory #2
ORUM	called by FORMIN and FORBIN to OR two values into one word
PAIN	sets address for variables, initializes DMA locations, initializes interrupts
UPIT	updates vector pointer and does output

ISR1

This routine services interrupts at the DEC LSI-11 when the interrupts are caused by DMA activity.

The DEC LSI-11 has only one interrupt level so there are no priorities. All interrupts, caused by DMA activity, cause the program counter to be set to the value in location 174 (octal). At the end of ISR1 code is an ASECT which places the starting address of ISR1 in location 174. The ASECT also places 200 (octal) in location 176. That value is assigned to the status register to mask interrupts. The source of interrupts can be external (CPU1) or internal (DMAL11). ISR1 services interrupts from CPU1 only.

An interrupt from CPU1 can mean that CPU1 wants the operator to set up the registers to receive data via DMA or that CPU1 has finished sending data via DMA. To tell the difference, one examines the byte count at location 177516. During initialization, subroutine PAIN sets the count at -1035 (decimal). When CPU2 sends data to CPU1, it uses subroutines DMAOUT. That routine sets the byte count to -1035 when the DMA activity is finished. Whenever CPU1 sends an interrupt to say it has finished sending data, ISR1 sets the byte count to -1035. All valid ways of changing the byte count also reset it to the expected value. If, on entry to ISR1, the byte count is not -1035, the interrupt was sent because CPU1 has finished sending data.

All this is necessary because one never knows how many bytes of data CPU1 is sending until it sends the interrupt saying it is finished.

When, on entry to ISR1, the byte count is -1035, CPU1 wants to send something via DMA. The three DMA registers must be set first, then CPU1 must be given a "go-ahead" interrupt. The three registers are locations 177512, 177514, 177516. In 177512 bit 15 is set to enable external interrupts. Bit 5 is cleared also to tell the DMAL11 it will be reading from CPU1.

The address of vector PASTIN where the data are to be stored is placed in 177514. The actual value is the address minus two. The DMAL11 increments the address by two as soon as it begins the DMA. The maximum possible byte count, -1035, is placed in 177516. This is the maximum because PASTIN has 516 words. The 1035 is made odd so that the byte count will be minus one if too much data are sent. The minus one will disable further interrupts. This will cause CPU2 to ignore CPU1's "all done" interrupt.

After all three registers are set, ISR1 clears variable IN1FLG. When the DMA is finished, ISR1 sets IN1FLG to one. Since this variable is used in many other routines to control processing, be careful with it.

After clearing IN1FLG, ISR1 tells CPU1 to begin its DMA. This is communicated to CPU1 by writing to location 177502 which causes an interrupt at CPU1. Then ISR1 waits for the "all-done" interrupt from CPU1.

DMAOUT

This routine sends vector to CPU1 via DMA.

Vector PASTIN must contain the data to be sent to CPU1. The first word must contain the identification code that tells CPU1 why it is receiving data. Word two has various meanings. Word three must contain a count of the words in PASTIN starting with word four. This is the count of words to be transmitted in addition to words 1, 2, 3. Figure 19 shows the PASTIN formats.

Subroutine DMAOUT uses the word count in word 3 of PASTIN to compute the byte count for transmission. The byte count = 2* (PASTIN (3) + 3) in FORTRAN notation. The byte count is made negative and assigned to location 177516. The DMAL11 board increments that location by two every time a word is sent. DMAOUT monitors that location. When it contains zero, CPU1 has taken all the data.

Location 177512 is the control/status word for the DMAL11 board. It is used to enable/disable interrupts at the DMAL11 and to tell the board if it should read from or write to the external device (CPU1). For writing in DMAOUT, set bit 5 in location 177512. This clears bits 14 and 15 and disables interrupts.

Location 177514 contains the starting address minus two for the vector to be transmitted. The DMAL11 increments the address by two before the first word is sent; therefore, the starting address must be diminished by two.

1. STANDARD FORMAT FOR CPU2 SENDING TO CPU1:

ID CODE	MEMORY ADDRESS	NUMBER OF WORDS FOLLOWING THIS ONE	DATA

2. FORMAT FOR ID CODE 2 TO CPU1 (WRITE MEMORY TWO):

ID CODE	X ADDRESS	NUMBER OF WORDS FOLLOWING THIS ONE	Y ADDRESS	DATA

3. FORMAT FOR ID CODE 6 TO CPU1 (READ MEMORY TWO)

ID CODE	X ADDRESS	2	Y ADDRESS	LENGTH TO SEND BACK

4. FORMAT FOR CPU1 SENDING TO CPU2:

ID CODE	NUMBER OF WORDS FOLLOWING	DATA

5. FORMAT FOR CPU1 SENDING MEMORY 1 CONTENTS TO CPU2:

ID CODE	NUMBER OF WORDS FOLLOWING	START ADDRESS	DATA

Figure 19. PASTIN Formats

Location 177500 is used to send an interrupt to CPU1 whenever anything is written to location 177500.

After CPU1 takes all the data, DMAOUT resets the byte count to -1035. This value must be there for correct operation in subroutine ISR1. Next DMAOUT enables interrupts from CPU1 at the DMAL11 by setting bit 15 at location 177512. Then bit 7 is set in the LSI-11's status word to enable interrupt there.

Memory Tests

Subroutine MEMEX1 tests memory #1. Subroutine MEMEX2 tests memory #2. The test is the moving inversions test described in Computer Design, May 1976, pages 169-173. There are two deviations from that description. The carry bit is not added to the address to affect varying address sequences. The other deviation is in testing memory #2. Entire rows of memory are tested at once rather than individual words. All of memory #2 cannot be addressed directly with only a 16-bit word. So X, Y coordinates are used. To test the memory, the value of X is set and Y is varied. The result is that the moving inversion test is performed on 512 "words," each containing 512 x 16 bits. Within a row, the bit under test is the same in all words at one time.

PAIN, PASTIN, DMAFLG, IN1FLG

Subroutine PAIN links FORTRAN common blocks with assembly language subroutines. Its primary purpose is to define PASTIN, DMAFLG, and IN1FLG for the assembly language routines. These are declared global variables. Since they contain the address of the corresponding variables in the FORTRAN routines, any reference to the value of the variable must be by deferred addressing.

DMAFLG was intended for use with subroutine ISR1 when internal interrupts were allowed. Since those interrupts are no longer being used (see ISR1), DMAFLG has no function at this time.

IN1FLG can be assigned the values zero or one. It is one when CPU1 sends data to CPU2. Subroutine ISR1 clears IN1FLG when the DMA registers are set just before sending the "go-ahead" interrupt to CPU1. When CPU2 receives the "all done" interrupt, part of the processing is to set IN1FLG to one. In the FORTRAN main routine for normal processing is a check on the value of IN1FLG. If it is one, control is transferred to subroutine CPU1 INPUT which sets flags to transfer data PASTIN to OBJECTS. Then IN1FLG is set to zero by CPU1 INPUT.

PASTIN is used to send and receive data from CPU1. It is 516 words long. This length was chosen to accommodate an entire row of values for memory #2 plus the four control words associated with that data. Normally there are three control words, but there are four for references to memory #2. The control words are as shown in the table of PASTIN FORMATS. The ID codes are explained in Table 6 and Table 7.

CPU2 SOFTWARE ABSTRACTS

The software abstracts that follow are of routines developed on the DEC LSI 11/2. Some are repeated in this section because it describes all the software developed in CPU2. Table 8 presents the five general categories of software developed in CPU2. These are diagnostics, interfacing with CPU1, interframe analysis, cueing, and other routines which don't fall into the first four categories. Following the abstracts for the software, a linkage table (Table 9) shows all the routines used.

TABLE 6. ID CODES FROM CPU2 TO CPU1

ID Code	Meaning	Status
0	Perform interface protocol test.	not functional
1	Put data in memory #1	functional
2	Put data in memory #2.	functional
3	Data has been put in memories #1 and #2 so execute with it (to be used in simulation).	not functional
4	Enter an idle loop while memory is being loaded.	not functional
5	Send memory #1 contents.	functional
6	Send memory #2 contents.	functional
7	Send CPU1 register contents.	functional
8	Prepare to write in memory #2.	functional
9	Terminate writing in memory #2.	functional
10	Grab a frame, send the object features, then "do nothing."	not functional
11	Real time	not functional

TABLE 7. ID CODES FROM CPU1 TO CPU2

ID Code	Meaning	Status
0	Feature data for objects	functional
1	Debugging message	not functional
2	Error message	not functional
3	Register contents	functional
4	Memory #1 contents	functional
5	Memory #2 contents	functional

TABLE 8. CPU2 SOFTWARE

Diagnostics	Interface with CPU1 Including Interrupt Servicing	Interframe Analysis	Cueing	Other
bitest*	cpulin	costit	aacan*	dmpstat
ckdspl	dmacode	elect	aalnch*	initia
disptst	dmaout*	huh	apc*	opinpu
d2mlwi	getum	inter	cueit	simula
d2m2dt	isrer	pairum	erase	simul2
dumpml	isrl*	pick	gbpchk*	super
dumpm2	rtisr*		nontarg*	train
ereprt	Macros*:		nothng*	pain*
erepr2	restor		puton	
forbin	saveum			
formln	erp		tank*	
lshift*			truck*	
memex1*			uit*	
memex2*			Macros*:	
memtel			addx	
memte2			addy	
orum*			draw	
rows*			hedr	
upit			whats	
Macros*:				
memer				
readit				
read2				
ckvals				
kludge				

*assembly language, all others are in FORTRAN

Diagnostics

- BITEST --Evaluates the bit being tested when memory #2 is being tested.
- CKDSPL --Calls DISPTST to test the display by checking the graphics plane bits and drawing the symbols.
- DISPTST --Tests the graphics components. Calls ROWS to set all bits to one, then reads all bits one at a time to verify. Calls ROWS to set all bits to zero, then verifies. Next all symbols in the symbol library are displayed one at a time.
- D2M1WI --Used to put data in any memory #1 location. Data can come from disk files or can be entered at the keyboard.
- D2M2DT --Used to put data in memory #2 locations. Data can come from disk files or can be entered at the keyboard.
- DUMPM1 --Prints the contents of any memory #1 locations or CPU1 registers.
- DUMPM2 --Prints the contents of any memory #2 locations.
- EREPRT --Prints error messages for errors discovered in testing memory #1.
- EREPR2 --Prints error messages for errors discovered in testing memory #2.

Diagnostics (cont.)

- FORBIN --Accepts data from the keyboard, formats the data as bin data would appear in memory #1, stores data on disk.
- FORMLN --Accepts data from the keyboard, stores it on disk so that later it can be transferred to CPU1 memories. There is no special formatting of the data.
- LSHIFT --FORTRAN callable routine which performs a left shift of the value passed in.
- MEMEX1 --Tests memories #1 and #2 using a moving inversions
- MEMEX2 --test. The procedure has been reported by J. Henk de Jonge and Andre J. Smulders in Computer Design, May 1976, "*Moving Inversions* Test Pattern is Thorough Yet Speedy." The procedure inverts the data of each address sequentially, thus creating an access time by the jump from one address to another which contains different information. Read/write/read operations are performed with both forward and backward address sequences.
- MEMTE1 --Sets up the range of addresses to be tested in memory #1.
- MEMTE2 --Sets up the range of addresses to be tested in memory #2.
- ORUM --FORTRAN callable routine which performs an OR operation on the arguments passed to it.

Diagnostics (cont.)

- ROWS --Used to set all graphics plane bits to a specified value. They are set a row at a time by writing vectors of length 128, four per row. It is assumed there are 512 rows with 512 bits in each row.
- UPIT --Used by FORBIN to manage pointer variables and disk I/O during construction of a file of bin data.
- MEMER --Used by MEMEX1 to set up the call to EREPRT.
- READIT --Used by MEMEX1 to set up the call to DMAOUT to read from memory #1.
- READ2 --Used by MEMEX2 to set up the call to DMAOUT to read from memory #2.
- CKVALS --Used by MEMEX2 to compare the values returned from memory #2 with those sent to memory #2. Any differences are reported by calling EREPR2.
- KLUDGE --Used by MEMEX2 to send code 8 or 9 to CPU1 so it knows a write action is being started or terminated, respectively.

Interface With CPU1

- CPU1IN --Decides what to do with the data passed to CPU2 from CPU1. When the input is a bunch of feature sets from a frame during real time operation, flag INTRFLG is set so SUPER can call INTER, the interframe analysis entry routine. If the data passed in are feature sets during a training session, flag TRANFLG is set so SUPER can call TRAIN. Error codes and diagnostic messages will be printed on the line printer when the data passed are of that type.
- DMACODE--Passes a control word to CPU1 from CPU2. The control word tells CPU1 what procedure is to be followed. The word to be sent is passed in to DMACODE. Then DMACODE concerns itself with all the interface protocol needed to send the message by calling DMAOUT.
- DMAOUT --Sends messages to CPU1 via DMA, sets up the memory locations used to invoke DMA, monitors the transfers, manipulates interrupts.
- GETUM --Used in interfacing; reads the object features passed to CPU2 from CPU1. The features were stored in a temporary buffer when CPU1 interrupted CPU2. GETUM copies the features to a permanent buffer.
- ISRER --Prints error messages generated by the interrupt service routines ISR1 and RTISR.

Interface With CPU1 (cont.)

- ISR1 --The interrupt service routine for use with diagnostic routines.
- RTISR --The interrupt service routine for use with SUPER.
- RESTOR --When the interrupt service routine, RTISR, is about to return control to the interrupted procedure, macro RESTOR restores all the register contents.
- SAVEUM --Stores all the register contents on entry to the interrupt service routine.
- ERP --Used by both interrupt service routines to set up the call to ISRER.

Interframe Analysis

- COSTIT --Used in interframe analysis; computes the cost of claiming that an object in frame N is the same as an object in frame N-1. The costs are computed for all possible pairings of objects in the two frames. Since the cost is the sum of the absolute differences between the object features, the most similar objects have the lowest cost.

Interframe Analysis (cont.)

- ELECT --Used in interframe analysis to decide what classification should be assigned to an object; calls the cueing routine to cue the object. ELECT counts the frames in which the object occurs and the classification assigned to the object in each frame. The classification assigned is the one which occurs most often in the frames queried.
- HUH --Used in interframe analysis routines to compute frame identifiers for use as subscripts in other subroutines. The identifiers are computed relative to the current frame identifier.
- INTER --Controls interframe analysis.
- PAIRUM --Used in the interframe analysis procedure. Subroutine PICK links objects in frame N with objects in frame N-1. If objects in frame N cannot be matched with objects in frame N-1, subroutine PAIRUM tries to match those objects in frame N-2. So PAIRUM links objects in frame N with objects in frame N-2.
- PICK --Used in interframe analysis to decide which objects in frame N should be matched with which objects in frame N-1. The pointers that link objects across frames are established by PICK.

Cueing

- AACAN --Draws a letter C superimposed on the object to be cued. See TANK.
- AALNCH --Draws a letter M superimposed on the object to be cued. See TANK.
- APC --Draws a letter A superimposed on the object to be cued. See TANK.
- CUEIT --Subroutine CUEIT is called to write a target symbol in the graphics memory for display. The calling routine passes the target coordinates, extent, and classification code. These data are stored in object queues so we know later what was cued and when; then we can erase the cue when we want to. Every time CUEIT is called, the frame identifier is checked to see if the target to be cued is the first one in a new frame. If it is the first one, then CUEIT calls ERASE to delete the oldest cues still displayed. The term frame means the interval between receiving one set of objects, to be cued, from CPU1 and receiving another set. Frame 1 begins when a set of objects to be cued is received. Frame 2 begins when CPU1 sends the next set of objects.

Cueing (cont.)

- ERASE --Subroutine ERASE resets the bits in graphics memory for symbols that have been displayed for a number of frames equal to the value of variable LIFE. ERASE does this by computing the difference between the identifier for the frame when the symbol was produced and the identifier for the current frame. That is the age of the cue. When that age is equal to or greater than LIFE, the symbol generation routine is called with the set/reset bit set to zero. This is assigned to the same bits that were set when the symbol was first produced. Notice that if an older and younger symbol overlap, the overlapped part of the younger symbol will be erased when the older symbol is erased.
- GBPCHK --Reads bits in the graphics bit plane and compares them with the expected value; returns an error code if the bit is not set correctly.
- NONTAR --Draws a letter O.
- NOTHNG --Draws an X across the upper left quarter of the screen. Called during training when the frame analyzed has no targets to cue. The X is erased as soon as a frame is found to have targets to be cued.
- PUTON --Interacts with the operator to draw a letter T anywhere on the display screen.

Cueing (cont.)

- TANK --Draws a letter T superimposed on the object to be cued. The size of the letter will be proportional to the size of the object. Actually, the long axis of the cue will be the same length as the longest dimension of the object. The symbol will retain a 5/7 ratio between width and height. The symbol will be superimposed rather than offset because it is easier to superimpose it. Later, if we want to offset, we can change software to make it happen. The arguments passed to TANK are expected to be the graphics memory coordinates of the upper left corner of a box and the limiting rectangle surrounding the symbol to be produced. The object extent, that is, the dimensions of the box, are to be expressed as the number of bits to set in the graphics memory.

- TRUCK --Draws a letter W superimposed on the object to be cued. See TANK.

- UIT --Draws a letter U superimposed on the object to be cued if it is an unidentified target. See TANK for more details of symbol generation.

- ADDX --A macro which adds its argument to the initial value of X, then gives the sum to the symbol generator.

Cueing (cont.)

- ADDY --A macro which adds its argument to the initial value of Y, then gives the sum to the symbol generator.
- DRAW --A macro which draws a line according to the arguments passed to it. Waits 53.9 microseconds for the symbol generator to complete each vector. Timing is based on a worst case with vector length 32.
- HEDR --A macro that standardizes symbol subroutine entry protocol. An error check is done on the number of arguments. The mode command word is set. And the upper left corner X and Y coordinates are set in the symbol generator.
- WHATS --A macro used to define nonvarying parts of command words to be passed to the symbol generator. The nonvarying parts are the vector number, vector direction, and write bit.

Other

- DMPSTAT --Used at the end of a training session. Statistics collected for all objects detected in all frames during a training session are saved in arrays. At the end of the training session the statistics can be printed by entering execution code 4 when the system asks for it.

Other (cont.)

- INITIA --Assigns initial values to variables and vectors.
- OPINPU --Used by the supervisory routine to interact with the operator. Queries or directives are printed and input is accepted by OPINPU. The input is verified as being at least valid, if not correct, before it is passed to other routines for use.
- SIMULA --Controls system simulation during checkout with test data input from floppy disk.
- SIMUL2 --Used to cue objects passed from CPU1 without going through interframe analysis. Demonstrates the interface is working and can be used to verify CPU1 procedures.
- SUPER --The main program which controls the whole system. SUPER contains the idle loop where control resides until there is a command from the operator to do something or until CPU1 sends an interrupt. Flags are checked in this loop to decide what should be done in response to inputs from these sources.
- TRAIN --Cues objects passed to CPU2 from CPU1 during training sessions. Each object is cued with a non-descriptive cue. The operator inputs the range, aspect, and classification. The object is cued with that class cue; the operator verifies; then all object features and the class code are output to floppy disk.

Other (cont.)

- PAIN --Establishes access to variables in FORTRAN routines from assembly language routines.

TABLE 9. LINKAGE BETWEEN ROUTINES

Routine or Macro	Calls or Uses	Is Called By or Used By
Diagnostics		
bitest		erepr2
ckdspl		disptst
disptst	gbpchk, rows, tank, apc, truck aacan, aalnch, uit, nontar, nothng	ckdspl
d2mlwi	pain, dmaout, isrl	
d2m2dt	pain, dmaout, isrl	
dumpm1	pain, dmaout, isrl	
dumpm2	pain, dmaout, isrl	
ereprt		memex1, memer
erepr2	bitest	memex2, ckvals
forbin	upit, 1shift, orum	
formln		
1shift		forbin
memex1	memer, readit, dmaout, ereprt, isrl	memte1
memex2	read2, ckvals, kludge, dmaout, erepr2, isrl	memte2
memte1	pain, memex1	
memte2	pain, memex2	
orum		forbin
rows		disptst, initia
upit		forbin
memer	ereprt	memex1
readit	dmaout	memex1

TABLE 9. LINKAGE BETWEEN ROUTINES (continued)

Routine or Macro	Calls or Uses	Is Called By or Used By
Diagnostics continued		
read2	dmaout	memex2
ckvals	erepr2	memex2
kludge	dmaout	memex2
Interfacing		
cpulin		rtisr
dmacode	dmaout	simula, super
dmaout		d2m1w1, d2m2dt, dumpm1, dumpm2, memex1, memex2, readit, read2, kludge, dmacode
getum		rtisr
isrer		isr1, rtisr, erp
isr1	isrer, erp	d2m1wi, d2m2dt, dumpm1, dumpm2, memex1, memex2
rtisr	erp, saveum, restor, getum, isrer, cpulin	super
restor		rtisr
saveum		rtisr
erp	isrer	rtisr, isr1
Interframe Analysis		
costit		inter, pairum
elect	cueit	inter
huh		inter, pairum
inter	huh, cosit, pairum, elect	super, simula
pairum	huh, pick, costit	inter
pick		pairum

TABLE 9. LINKAGE BETWEEN ROUTINES (continued)

Routine or Macro	Calls or Uses	Is Called By or Used By
Cueing		
aacan	draw, addy, hedr	cueit, erase, disptst
aalnch	draw, hedr	cueit, erase, disptst
apc	hedr, draw, addy	cueit, erase, disptst
cueit	nothng, erase, tank, apc, truck, uit, nontar, aacan, aalnch	elect, train, simul2
erase	tank, apc, truck, uit, nontar aacan, aalnch	cueit, train
gbpchk		disptst
nontarg	hedr, draw	cueit, erase, disptst
nothng		cueit, erase, disptst, super, train, simul2
puton	tank	
tank	draw, hedr, addx	cueit, erase, disptst, puton
truck	hedr, draw	cueit, erase, disptst
uit	hedr, draw	cueit, erase, disptst
addx		tank
addy		aacan, apc
draw		aacan, aalnch, apc, nontar, tank, truck, uit
hedr		aacan, aalnch, apc, nontar, tank, truck, uit
whats		tank
Other		
dmpstat		super
initia	pain, rows	super

TABLE 9. LINKAGE BETWEEN ROUTINES (concluded)

Routine or Macro	Calls or Uses	Is Called By or Used By
pain		initia, memte2, memte1, dumpm2, dumpm1, d2m2dt, d2m1wi
opinpu		super, train
simula	dmacode, inter, simul2	super
simul2	nothng, cueit	simula
super	opinpu, initia, dmacode, nothng, train, inter, simula, dmpstat	
train	erase, nothng, cueit, opinpu	super

SECTION 4

PLANS FOR NEXT REPORTING PERIOD

Since much of the hardware is completed, our major effort will be to get the software operational in preparation for a flight test in February. The next report should be the final report which will review the PATS program and present an evaluation of the flight tests.

APPENDIX A

TEST PLAN FOR THE PROTOTYPE AUTOMATIC TARGET SCREENER

APPENDIX A

TEST PLAN FOR THE PROTOTYPE AUTOMATIC TARGET SCREENER

The following is a preliminary plan for testing the Prototype Automatic Target Screener. It is intended to be a starting point for determining just what should be measured and how it should be measured. This plan is not final and will be changed over the next few months after discussions between Honeywell and NV&EOL.

1.0 SCOPE

This document describes the test procedures to be used in testing the HG1056AA01 Prototype Automatic Target Screener. Some of the tests are predelivery and some to be performed at NV&EOL. The tests are designed to demonstrate PATS' performance.

2.0 APPLICABLE DOCUMENTS

The following documents form a part of the test procedure to the extent specified.

2.1 Military

1. Request for quote DAAK70-77-Q-1252, purchase description for automatic target screener.
2. NV&EOL contract, DAAK70-77-C-0248

2.2 Honeywell

2.2.1 Drawings--

10069490 Adaptive contrast enhancement AGC section	A1
10069486 CCD321A & drive #5	
10069840 DC restore and global gain bias control	A2
10067922 CCD 321A3 & drive #1	
10069841 Video switches and separator	A4
1006984 Sync and timing	A6
10069060 Background filter and bright threshold	A8

10068106	CCD32A3 and drive #4	
10069069	Edge and edge threshold	A10
10068104	CCD321A3 and drive #2	
10068105	CCD321A3 and drive #3	
10070192	Interval	A11
10070193	First level features	A12
10070196	Power monitor and CPU1/symbol	A14
10067925	Memory #2 bit plane	A21, A22, A23, A24
10069253	A/D converter	A26
10069251	Memory #2 control	A27
10070190	Symbol memory	A28
10070195	CPU1/CPU2 interface & program memory	A29
10070188	Input buffer and DMA controller	A31
10070187	CPU1 sequencer	A32
10070185	CPU1 processor and multiplexer	A34
10070189	Memory #1 constants	A36
10070186	Memory #1	A37
10070194	Writeable control store	

2.2.2 Documents--

1. At present time, no operations or maintenance manuals exist.

3.0 SYSTEM DEFINITION

The HG1056AA01 system to be tested consists of the following basic end items.

1. Prototype automatic target screener
2. LSI 11/2 computer with keyboard/printer and dual floppy disc

3. Intel microprocessor development system

4. Video camera for video source

5. LOHTADS FLIR

4.0 TEST REQUIREMENTS

4.1 General

This section describes the basic test requirements for conducting system tests on PATS. Areas included are:

Test responsibility
Test facilities and test equipment
Test conditions
Test discrepancy procedures
Test witnessing
Detailed test procedure
Test report

4.2 Test Responsibility

The tests will be conducted by personnel from both NV&EOL and Honeywell. Primary responsibility will be with Honeywell Research Center personnel during the ground testing and with NV&EOL personnel during the flight testing.

4.3 Test Facility

Besides the equipment mentioned in Section 3 (System Definition), the ground testing will require two video tape recorders and video data previously recorded with the FLIR to be used in flight test. The first video recorder will provide the raw video data; whereas, the second recorder will record the results. It would be desirable to have an 875 time base corrector used with the first recorder to provide a stable video source.

The flight test facilities including an aircraft (no low frequency vibration should be present) and range signal derivation will be the responsibility of NV&EOL. The equipment will be shock mounted in the aircraft. Honeywell personnel will be present to assist in the field installation. The targets to be viewed with the FLIR will be the same type as the training tape data.

The tests will be conducted under normal ambient conditions. Electrical power requirements are:

115V ± 10v, 400Hz, ≃ 400 watts (Aircraft power)

Fans for cooling will have to be changed depending upon the power source used.

In order to perform the image enhancement tests, an IR collimator and thermal source to generate various temperature targets will be required at NV&EOL.

4.4 Test Conditions

All measurements and tests should be made at standard ambient conditions. The system is designed using commercial rated parts and the integrated circuits are mounted in sockets. In order to fly the system, the integrated circuits have to be tacked down but may still be prone to coming out of their sockets because of vibration. The amount of vibration must be minimized.

4.5 Anomalous Test Conditions

In the event of an anomalous condition during the testing, the condition will be recorded. Necessary corrective action will then be taken to restore the affected hardware to its proper condition. Following the corrective action, the tests could resume in one of three ways:

1. Repeat all previously conducted tests
2. Repeat a portion of the previously conducted tests
3. Resume testing where the discrepancy occurred

Selection of one of these courses will depend upon the nature of the discrepancy and how it related to other tests to be conducted on PATS.

4.6 Test Witnessing

NV&EOL will be notified prior to the start of system testing at Honeywell. Three stages of testing will probably be done.

1. Preliminary ground testing at Honeywell
2. Ground testing at NV&EOL
3. Aircraft testing at a site chosen by NV&EOL

4.7 Test Procedure

4.7.1 General--Testing of the combined FLIR/PATS system consists of two parts: image enhancement tests and target screener tests. The image enhancement tests are performed in a laboratory environment on the ground and the target screener tests are performed on the ground followed by some tests in the air.

Performance goals for the target screener with image enhancement were listed in the RFP and are repeated here for reference purposes since they form the basis for the test procedures.

- Performance Goals
  - --Image Enhancements: The target screener shall be designed to perform the following image enhancements on the raw video for presentation to the observer. Provisions will be incorporated to permit switching from automatic to manual control.
  - --Automatic Gain Brightness Control: The automatic gain and brightness control shall adjust the overall gain (contrast) and brightness based on global or local area imagery characteristics. Local area control is preferred. The local control area should not exceed 1% (nominal) of the total displayed area. The MRT in automatic mode shall be maintained at no more than that obtained by manual control plus 10%. In the presence of noise this function may allow black or white saturation of the displayed video at points of peak noise. Any such saturation will be limited to 3% or less by area of the local control area. No artificial artifacting shall be introduced by this control.

--Performance Tests: Modified versions of the MRT shall be performed using standard 4-bar test patterns.

A. Manual vs Automatic Operation: Measure MRT with a trained observer manually setting the gain and brightness. This is $MRT_M(f)$. Then using the same display and observer, repeat the MRT measurement under automatic control. This is $MRT_A(f)$. The following limits shall apply:

$$MRT_A(F) < 1.1 \times MRT_M(f)$$

for: $O < F < F_R$

where F_R is the resolution limit frequency of the system.

B. Multiple Target Test: Using the MRT curve obtained under automatic control ($MRT_A(f)$, select a high temperature difference target (high f) and a low temperature difference target (low f) which are both just resolved by the system under automatic control. When the system is operating in automatic mode, both targets will be resolved when viewed simultaneously in the same field of view and separated by a distance equivalent to that subtended by the local control area.

--Synthetic DC restoration: This enhancement shall restore to the displayed image a proportion of the DC or background component of the scene and eliminate the streaking or overshoot effects commonly associated with AC coupling a scene with high signal excursions.

o Performance Tests

--Normalized Mean Square Error: The performance of the synthetic DC restoration shall be determined by measuring the normalized

mean square error (NMSE) between the relative intensities in the scene (target) and the corresponding relative intensities in the display with automatic gain and brightness controls set to manual. The measure shall be defined as follows:

$$\text{NMSE} = \frac{\iint \left[\left[RI_o\ (x,\ y) - RI_I\ (x,\ y)\right]\right]^2 dA}{\iint \left[\left[RI_I\ (x,\ y) - \overline{RI_I}\right]\right]^2 dA}$$

where:

$RI_I\ (x,\ y)$ = Relative intensity of input scene at point x, y

$$= \frac{I_I\ (x,\ y) - \overline{I_I}}{I_{IMAX} - I_{IMIN}}$$

and:

$I_I\ (x,\ y)$ = Intensity (arbitrary units) of input scene at point x, y.

$\overline{I_I}$ = Average of I (x, y) over whole input image.

I_{IMIN} = Noiseless (time averaged) intensity (same units) of input scene at its point of minimum intensity.

I_{IMAX} = Same as I_{IMIN} but measured at point of maximum intensity.

$RI_o\ (x,\ y)$ = Relative intensity of output image defined analogously to $RI_I\ (x,\ y)$.

$\overline{RI_I}$ = Average of $RI_I\ (x,\ y)$ over input image.

The synthetic DC restoration should produce NMSE 20 percent when using the following test patterns:

A. Horizontal black and white (hot & cold) bars (that is, along scan lines) which fill about one third of the length of a scan line. The background shall be a third intermediate temperature.

B. Diagonal Test Target. A 45° diagonally divided target which is half hot, half cold and whose overall size falls about half the length of a scan line. The background shall be a third intermediate temperature.

--Target Screening: The target screener shall operate on the video information available from the imager to extract, classify, and cue targets for the human observer.

--Number of Candidates: The design shall be capable of simultaneously processing multiple candidate targets as they are extracted from the video. The screener fabricated for this program shall be capable of simultaneously processing at least 10 candidates, per frame; however, the design shall be expandable.

--Classes: The target screener shall be capable of classifying extracted candidate targets into one of five classes. All classes will be of fixed shape.

The classes shall be oblique aspect views of each of the following targets:

2 1/2-ton truck
Tank
Armored personnel carrier
Track-mounted radar-controlled anti-aircraft cannon
Track-mounted anti-aircraft missile launcher

The screener shall be capable of this classification over ranges from the 90 percent human detection range to one third of that range without benefit of range data. (Range has been added and can be used.)

--Recognition Capability: At ranges where the probability of human detection is 90 percent, the probability of recognition of targets by the screener shall be 80 percent. Misclassification of a target (that is, tank as a truck) shall be considered non-recognition.

--False Alarms: Classification of any non-target clutter or nuisances as targets shall be considered a false alarm. The average false alarm rate shall not exceed 1 per frame processed.

--Priority Target Search: A separate mode where the classes of targets sought are restricted to only two shall be included. In this mode only radar controlled anti-aircraft cannon and anti-aircraft missile launchers will be recognized and presented to the observer. A key objective will be extremely low false alarm rates. False alarm rates of one per 200 frames should be goal.

--Processing Update: The automatic target screener shall process the search frame sufficiently fast to update at a 10-frame-per-second rate.

--Cueing Output: A visual cue which unambiguously marks the target recognized by the screener shall be added to the video information. This mark shall indicate the class of target.

--Construction: The target screener shall be constructed using good commercial practice and considering that this is a breadboard unit. Connection to the GFE imager shall be without modification to the existing printed circuit cards. The screener may be fabricated in a separate case; however, integration or attachment to the MODFLIR would be desirable.

--Display Output: The video output shall be composite video suitable for driving a 525 or 875 line monitor.

The image enhancement tests will be made at NV&EOL; the ground testing will be done initially at Honeywell and then at NV&EOL, and flight testing will be done at NV&EOL at a site chosen by NV&EOL.

4.7.2 Image Enhancement Testing--

4.7.2.1 Manual vs Automatic Control--Operation of the FLIR under normal conditions will be tested by having the PATS hardware image enhancement switch set to manual. Adjustment of gain and brightness will be observed. When the PATS is switched to auto, the gain and bias controls on the monitor should have no effect.

4.7.2.2 Minimum Resolvable Temperature (MRT) Tests-- The MRT shall be done for manual control and then under automatic control. The ratio of the two shall be determined and noted. The amount of saturation (black and white) should be noted in the presence of noise. Specific requirements relating to these tests were described in Section 4.7.1 under the goals of the image enhancement performance tests.

4.7.2.3 DC Restore--Normalized mean square error measurements will be made in accordance with Section 4.7.1 and noted. Two test patterns are required. These are horizontal black and white (hot and cold) which fill about one third the length of a scan line with an intermediate third temperature as background. The second test target is a 45° diagonally divided target which is half hot and half cold and overall fills half the length of the scan line. The rest of the scene is background with a third intermediate temperature.

4.7.3 Target Screener Tests--Section 4.7.1 describes the target screener goals. However, range will be tested as part of this program. Figure A-1 shows the anticipated configuration for the ground testing. The DEC LSI 11/2 will be available for additional training as necessary and should not actually be in the test. The Intel MDS system with a writable control store will be used for microcode program storage prior to the use of proms in the flight tests.

The video disc would enable one to grab selected frames from a video tape and then count the number of objects processed and view their classification for extended periods of time. The disc must be capable of handling 875 line video.

4.7.3.1 Target Screener Tests--The target screener tests should be broken into three basic phases: the training or gathering of data necessary to do detection and classification, ground testing of the PATS system using selected video tape segments, and a limited flight test program. The testing should be done in this order in order to minimize risk to the hardware and achieve the best performance possible in an actual flight environment.

The training phase is really not a test phase, but it must be done before any other testing can be done. The video data required for this phase must be similar to the flight tests data and must be taken with the FLIR the system will be tested against. Range data to the targets should be recorded on the audio channel as well as the number and type of targets in the video. Additional comments should be made as to where the observer first sees the targets and when he recognizes the targets. The observer should not have a priori knowledge about the targets on their location. This will help in the evaluation of PATS.

GROUND SYSTEM TEST SET-UP

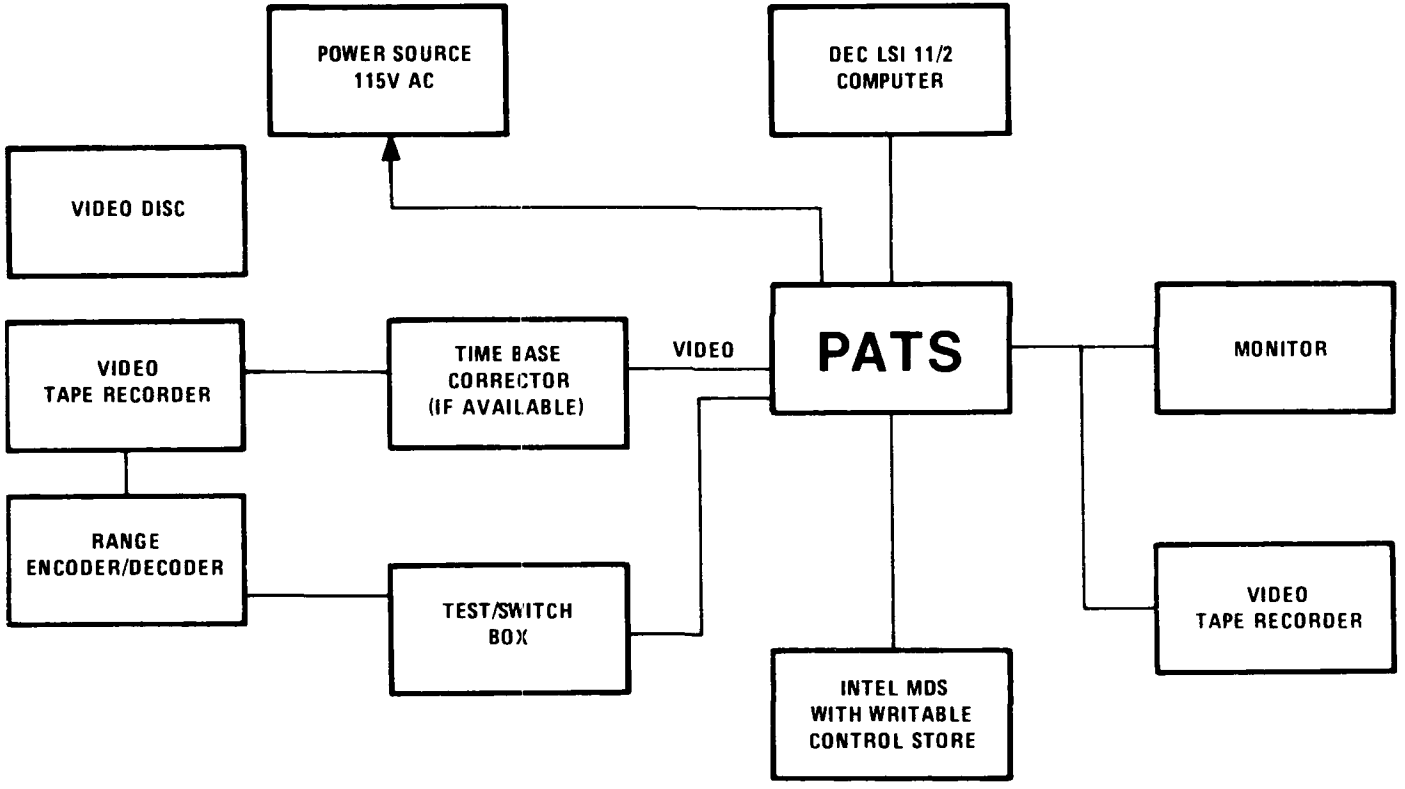


Figure A-1. Ground System Test Set-Up

It is anticipated that the general scenario would include all the target classes and several of each target class. The system cannot be trained without an adequate training data base. The scenarios should include pop-up type mission overflights, flights directly toward the target starting at a given range, and altitude and circling flight where altitude and range may either be fixed or varied at some predetermined rate.

4.7.3.2 Functional Tests--Once the system is trained, some preliminary functional tests can be performed. There are few switches on the PATS hardware. The following general checks can be made on the system.

Manual/Auto Range--Determines whether the range data comes from thumb-wheel switches or from the transponder.

Power Indicator--Indicates whether power has been applied to the system.

Priority--Determines whether PATS will operate in priority mode for locating anti-aircraft cannon and anti-aircraft missile launcher.

Switches relating to image enhancement and video control were described in Section 4.7.2.

4.7.4 Target Screener Ground Tests

4.7.4.1 General--

1. The recognition tests should consist of scoring the performance of PATS in recognizing the specified targets.

2. The detection probability will be determined as part of the testing. Classification of a target into the wrong class will still be applicable to the detection.

3. The input data will come from video tape data which should be time-base corrected.

4. Applicable input imagery and ground truth definition of the imagery will be mutually agreed upon by NV&EOL and Honeywell prior to conducting the tests. The ground tests will consist of inputting segments of imagery which have been mutually agreed upon as acceptable data. When the PATS system recognizes a target, it will put a letter symbol under the target on the video imagery. The performance will be based upon the presence or absence of these symbols on the video.

5. In order to evaluate the performance, it may be necessary to put the tape recorder which has recorded the symbol in pause to determine which target was recognized.

6. The first three quarterly reports presented the general algorithms used and the general type of performance that was obtained with a selected set of frames from previously provided NV&EOL imagery. The results were based upon a subset of the possible target aspect and elevation angle. Also there was only a subset of the targets evaluated because of lack of data for the other classes.

4.7.4.2 Procedure--The following basic procedure should be used for each selected segment of input imager.

1. Record the mission/recording information applicable to the specific reel of video tape to be used.

2. Observe settings of the PATS system with regard to data to PATS, 11-line rate and range for the desired test. Two or more runs will probably be made with each segment.

   a) Video without range, without enhancement

   b) Video with range, without enhancement

   c) Video without range, with enhancement

   d) Video with range with enhancement

3. The video output which would normally be displayed on the FLIR monitor will be recorded along with the audio channels from the input type recorder.

4. The range at which targets are recognized will be recorded for each segment. The number of correct classifications, incorrect classifications and false alarms will be recorded.

5. The get new frame pulse should be monitored. The average time between pulses should be used to determine the average processing rate of PATS.

6. The goals for PATS as stated in the RFP are listed below:

   a) Capable of simultaneously processing 10 candidates per frame.

   b) Capable of classifying into one of five classes.

   c) Capable of classification over ranges of 90 percent human detection range to one third of that range without benefit of range data. Range data will be checked to see if it improves the classification.

d) Recognition capability shall be 80 percent at a range where a human can detect 90 percent. This detection range should be noted on the audio channel by the pilot during the training data gathering in order to determine the range over which the target screener is to be tested.

e) Average false alarm shall not exceed one per frame. False alarm is defined as calling a piece of clutter a target. Misrecognition of targets will not be counted as a false alarm but rather as decreased recognition; that is, the target has been detected.

f) For priority mode (presently no data available) the goal was one false alarm per 200 frames. This particular phase cannot be tested since no data are available.

4.7.4.3 Construction--The target screener was initially constructed using the commercial practice of wire wrap boards for most of the digital hardware and point to point wiring on the rest of the digital and the analog boards. Because the system is mounted in sockets, it could be subject to vibration. Because of this ground tests should be done to establish performance before the air tests are conducted.

4.7.5 Flight Tests--

4.7.5.1 General--The target screener will be installed in the helicopter by Honeywell and NV&EOL personnel. The PATS system will have been optimized for best ground results before being installed in the helicopter. Shock mounters are a necessity. The general configuration for the aircraft is shown in Figure A-2. The test/switch box will be accessible to the pilot

AIRCRAFT TEST SET UP

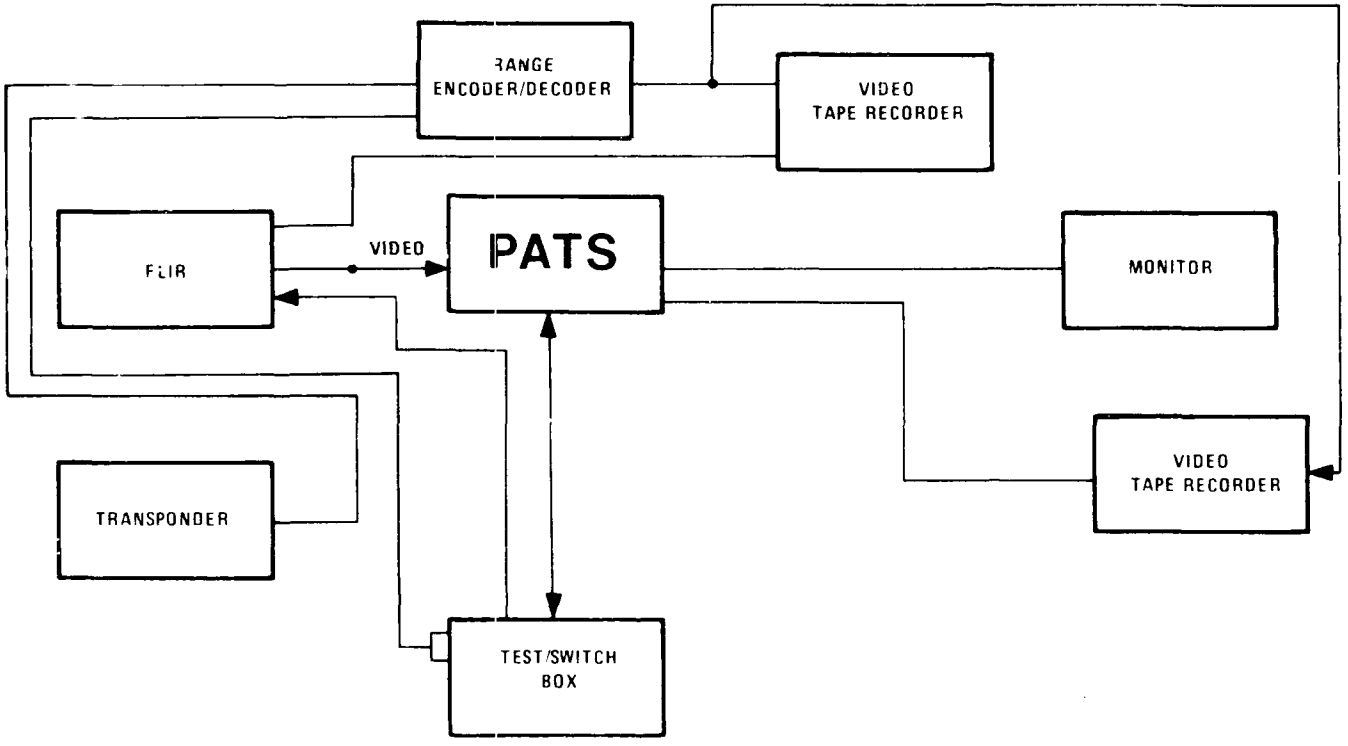


Figure A-2. Aircraft Test Set-Up

AD-A095 596 HONEYWELL SYSTEMS AND RESEARCH CENTER MINNEAPOLIS MN F/G 17/5
PROTOTYPE AUTOMATIC TARGET SCREENER.(U)
MAY 80 R C FITCH, D V SERREYN, T G KOPET DAAK70-77-C-0248
UNCLASSIFIED 80SRC73 NL

END
DATE FILMED
3-81
DTIC

and/or passenger/copilot seats. This will make all controls except power-on easily accessible during the flight tests.

4.7.5.2 Ground Coordination--The ground targets shall be set up by the NV&EOL personnel at the site chosen by NV&EOL. The targets may be either stationary or moving. Range should be determined to the center of the FOV for the specific target with the transponder.

4.7.5.3 Flights--Various missions will be flown with PATS. The video data that is input to PATS will be recorded. This data may be operated on by the DC restore and global gain/bias control circutry in PATS depending upon the switch settings.

Flights with various altitudes will be flown against a given set of targets. It is anticipated that the targets will be similar to those used during the training data gathering. These flight paths should be determined prior to flight tests.

4.7.5.4 Performance Evaluation--The data tapes gathered during the flight will be evaluated on the ground. Scoring will take place in the same fashion as the scoring of the ground tests. The evaluation procedure will be defined prior to ground or flight tests.

4.8 Test Report

The results of the testing of PATS will be included as part of the final technical report generated for the PATS program. This report should be submitted 30-60 days after the tests are completed.

APPENDIX B

PAPER ABOUT
PROTOTYPE AUTOMATIC TARGET SCREENER

APPENDIX B

PAPER ABOUT PROTOTYPE AUTOMATIC TARGET SCREENER

The following paper was presented at the U. S. Army Missile Command Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance held 19-20 November 1979.

PROTOTYPE AUTOMATIC TARGET SCREENER*
GOALS AND IMPLEMENTATION

By

D. E. Soland, D. V. Serreyn, R. C. Fitch, T. G. Kopet
G. O. Prom, R. C. Reitan, M. O. Schroeder

HONEYWELL, INC.
Systems and Research Center
2600 Ridgway Parkway
Minneapolis, Minnesota 55413

ABSTRACT

The Prototype Automatic Target Screener (PATS) is being developed at Honeywell under contract with the Army Night Vision and Electro-Optical Lab. The system consists of hardware for image enhancement to improve the imagery displayed to the operator of a FLIR and hardware and software for real time detection, recognition and cueing for selected tactical targets.

The PATS system will operate with standard 525 and 875 line TV formats. Decisions on target classification and location are updated every 1/10 second. The resultant decision is displayed by means of symbology overlays on the video display to the operator.

The hardware consists of twenty-two 6" x 9" boards featuring charge coupled devices to perform the high speed functions for image segmentation. It incorporates a bit-slice microprogrammable digital processor for classification as well as a bit plane structure frame memory. The hardware fits into a box slightly larger than an ATR box and dissipates approximately 200 watts. Each board is somewhat modular in function and boards of the same function could be easily substituted.

*The work leading to this paper was supported in part by the U.S. Army Night Vision and Electro-Optical Laboratory Contract DAAK70-77-C-0248.

INTRODUCTION

Under contract with the Army Night Vision and Electro-Optical Lab, Honeywell has simulated, designed, and is in the process of fabricating and testing a Prototype Automatic Target Screener (PATS). This program started in late 1977 and will culminate with ground and flight testing in early 1980. The system is designed to interface with a Common Module FLIR. The PATS system will automatically detect and recognize targets and cue the FLIR operator. This paper discusses the goals of the PATS program and then the actual implementation of the target screener. Additional information about the simulation were presented at the April 1979 SPIE Conference[1].

Image Enhancement Goals

Three image enhancement functions are to be provided as part of PATS. These are: (1) adaptive constrast enhancement, (2) DC restore for AC coupled detector systems, and (3) automatic global gain and bias controls. The performance goals are such that: (1) local area control should not exceed 1 percent of the total scene (∿2,500 pixels) and (2) the MRT degradation should be less than 10 percent. The synthetic DC restore will restore to the displayed image a proportion of the DC or background component of the scene and eliminate the streaking or overshoot effects commonly associated with AC coupling. This will be accomplished such that the normalized mean square error on two specified test patterns will be less than 20 percent. The two test patterns are alternating horizontal black and white bars and a black and white diagonal target.

Target Screener Goals

The target screener is designed to operate with any RS-343 standard 875 line video as well as RS-170 standard 525 line video. Both operate at a 60 Hz field rate. The specific FLIR system for testing will be a Lohtads Common Module FLIR. For image enhancement, the system must process every frame but the target screening function is required to process 10 frames/second or every third frame. For each processed frame a minimum of ten objects must be processed.

The target screener shall be capable of classifying extracted candidate targets into one of five classes. These five target classes are:

- 2½ ton truck
- tank
- armored personnel carrier

- track mounted radar-controlled anti-aircraft
- track mounted anti-aircraft missile launcher

The target classes can easily be changed through software. The recognition capability goal is at ranges where the probability of human detection is 90 percent; the probability of recognition by the screener shall be 80 percent. The target screener is to work from the 90 percent detection range to 1/3 that range. Originally, range measurements were excluded, but the PATS system is being modified to be tested with and without measured range. In Lohtads, range is measured by a laser.

The average false alarm rate shall not exceed one per frame processed normally. There is also a priority search mode where only the last two target classes are the targets to be cued. The objective with this mode will be lower false alarm rate. Specifically, the goal is one false alarm per 200 frames.

PATS System Design

The PATS system has several functions which it must perform on video data. These functions are shown in Figure 1. The first thing that is performed is image enhancement. This function is primarily for the displayed imagery but also may aid the target screening function. The rest of the functions shown in Figure 1 relate to the target screening function. Image segmentation must first be done to outline regions or objects of interest. Once the objects have been segmented, certain features must be measured which are used for initial recognition or classification within the frame. All objects in a frame are classified as to clutter or type of target.

The object classification for each frame is accumulated over a series of frames. When confidence is high enough that the decision is a specific target, a symbol indicating the classification is displayed. The sequential frame classification is called interframe analysis and is used to reduce the false alarm rate.

In the ensuing paragraphs of this paper we will discuss the implementation of these functions. Figure 2 shows the hardware configuration modules for each function.

The first module is the sync and timing generation. This module consists of two boards--one for sync separation and video switching and one for system timing. The sync separation and video switching is shown in Figure 3. The sync separator extracts the composite sync from the video. From the composite sync signal, basic sync signals such as vertical reset, field indica-

tor and horizontal sync are derived. Since the video is AC coupled, the video data must be black clamped.

Another function performed by this board is the multiplexing of video signals. Video to the monitor can either be raw video, enhanced video, analog test signals, or digital test signals. Similarly, the video to the target screening functions may be either enhanced video or raw video.

The second board is the system timing generator shown in Figure 4. This board produces sync signals and clocks that are synchronized with the incoming video. Two clocks are produced--a 455 clock and a 512 clock. The 455 clock is used for the analog CCD shift registers in PATS while the 512 clocks are used for the digital hardware. The signals generated by the board are commonly available in single chip form for 525 line commerical television but not for 875 line. For this reason, the function had to be built from MSI chips. The line rate must be manually set to agree with the FLIR configuration. For the 875 line format, we have 512 or 455 samples per 32 microseconds whereas with the 525 line rate we have 512 or 455 samples per 53 microseconds. The number of samples per line is considered sufficient for the current system requirements but can be increased if necessary.

The image enhancement function shown in Figure 2 consists of two boards which perform synthetic DC restoration, global gain and bias control and adaptive constrast enhancement. These are implemented with charge coupled devices and a microprocessor among other standard MSI and LSI parts. The particular functions and implementations are discussed in the references[2] and will not be repeated here.

The feature extraction function shown in Figure 2 consists of autothreshold hardware, interval and first level feature hardware. The autothreshold hardwave (Figure 5) consists of two analog processing boards which do intensity thresholding and edge derivation. Data is compared to a calculated adaptive threshold and a digital output is produced.

One of the functions performed by the autothreshold hardware is the generation of "hot" and "cold" signals. "Hot" data consists of those values above the background by a specific amount whereas "cold" is data below the background by a specified amount. The background filter is a two-dimensional low-pass recursive filter which operates at video rates. The threshold is computed from the video after the background estimate is subtracted. The threshold is based upon the variance of the video. The threshold value is multiplied by a predetermined constant to provide the video comparison. Exceedance of the threshold produces a logical true signal.

The second function performed by the autothreshold boards is the generation of edges for objects in the scene. The basic edge computation is a two-dimensional horizontal component. The edge threshold is based upon the scan line average of the analog edge. After the threshold computation and edge comparison, a logical edge signal is produced indicating the presence of an edge at that particular location in the video.

Digital signals from these two boards (edge, hot, and cold) is input to the interval boards shown in Figure 6. The PATS interval circuits include the implementation in bipolar TTL logic of a number of functions.

- Generation of an interval based upon previous, present, and next scan line edge, hot and cold signals.
- Validation of an interval as meeting certain practical constraints.
- Storage and generation of key interval-related data.
- Making interval data available to the processor memory and informing it that valid data is ready.

Interval generation is based upon the presence of a hot or cold signal in coincidence with an edge. Without the presence of an edge the data is invalid. Line delays are provided by digital shift registers and some of the logic is implemented in PROMS. As a result of this hardware the following first level feature data are stored in latches:

- Line number or Y position
- Number of intervals for each line
- Starting X position for each interval
- Width of each interval
- Background estimate at the start of each interval
- Sum of the intensities within each interval
- The bright count within each interval
- Indication of when the edge associated with the interval was located (start or end or both)
- Indication of interval as being either "hot" or "cold"

At the end of each line, the number of intervals and the Y position are transferred. At the end of each valid interval, data associated with that interval is transferred.

The next function in Figure 2 is the data memory. This memory stores up to ∿2,500 sets of interval data. This is a static RAM memory and data is transferred to it via DMA control. Data from the interval board are dumped into a FIFO shift register. A maximum of 21 intervals per line can be stored. This data is later transferred into a second FIFO and then finally into the data memory.

The data memory is part of the subsystem called CPU1. The CPU1 system architecture is shown in Figure 7. The basic processing unit consists of a high speed microprogrammed data processing register and arithmetic logic unit (4-2903), a 16 x 16 high speed multiplier, a microprogram sequencer and an addressing register and arithmetic logic unit (4-2901). The microprogram is stored in high speed RAMs during checkout and debug stage and will be transferred to PROMs for testing.

The CPU1 is interfaced to the various memories via the data and address bus and also to an external computer for debug and checkout. The second computer, A DEC LSI 11/2, is not used during the actual operational mode of the PATS hardware. It is, however, used as part of the training of the hardware.

Both CPU1 and CPU2 have access to the symbol generation hardware shown in Figure 8. Only the CPU2 connection is shown. Symbols are generated by writing vectors into a graphics bit plane. The data is read out of the bit plane at video rates. This is accomplished by using a parallel to serial converter on the output and addressing only every eight pixels. The displayed video is replaced by the symbol. The symbol size and shape is programmable.

Once a target is detected the necessary data for symbol generation are X, Y position within the frame, target classification and target size. As the target moves, the displayed symbol is erased and a new symbol is generated. As the target gets larger, so does the symbol.

In the lower portion of Figure 2 is shown an A/D conversion block. There are two A/D converters in the PATS converter unit shown in Figure 9. One A/D determines the digital value of the background estimate at the beginning of each interval. The second A/D is used for digitizing the entire video frame. Also included on the board are provisions for testing the A/D and testing the frame store memory. Both converters are 8 bit high speed TRW A/D converter chips.

An additional item included as part of the A/D converter unit is a 12 bit summer. At the start of an interval, the summer is cleared, and data is then summed over the entire width of the interval. This gives the sum of intensities from which the average intensity can be calculated. Only the eight most significant bits of the sum are transferred to CPU1 for processing.

The digital data coming from the second A/D converter are transferred to the Memory #2 (frame store) at video rates. This memory is made up of memory bit planes as shown in Figure 10. Each bit from the A/D converter goes to a designated bit plane. The bits are shifted into a serial shift register and on every eighth pixel, data is transferred into the actual memory chips. Eight 16K x 1 dynamic RAMS with access times of 375 nsec are used for the memory. This data can be randomly accessed by CPU1 for calculations necessary to do the recognition and classification of targets.

Software

Much of the processing for detection and recognition of targets is done in CPU1. In Figure 11, the software functions are shown. The software sequence is:

- Bin matching
- Median filter
- Object feature generation for clutter removal
- Clutter recognition
- Recognition features
- Classification
- Interframe analysis
- Symbol generation

All processing in CPU1 is completed in 0.1 second.

Bin matching associates the one-dimensional intervals characterized by the interval features into sets of intervals which determine two-dimensional objects. That is, CPU1 reads interval data from the memory, reorders them and then writes them back. The matches of intervals on a scan line by scan line basis is primarily determined by the location of each interval within its scan line.

Once a bin is complete, i.e., no more intervals match, CPU1 will smooth the boundaries of each bin using a one-dimensional median filter of width five. The values which are input to the filters are the endpoints of the intervals making up each bin. A separate filtering operation is done on the left and right hand edges of each object.

Object features are computed by CPU1 on the median filtered bins in a hierarchial fashion. That is, less expensive features are computed initially to do preliminary clutter screening and more expensive features are computed on the unrejected objects in order to do additional clutter screening and object recognition.

The classification algorithm is the k-nearest neighbor. The recognition classifier puts each active object bin into one of five target categories using moment features for that object and stores that classification together with the object size and location. The data is then processed by the interframe analysis.

The interframe analysis associates objects between frames. Using a Baye's decision, the object is classified and a symbol is generated. The symbol generated is directly related to the classification derived.

The second CPU can do interframe analysis and symbol generation. This allows one to check out the object matching. CPU1 will be doing this in the hardware to be delivered. CPU2 is still needed for training and diagnostics. Figure 12 shows the functions that CPU2 can provide. It has the capability of dumping data to or reading data from the two memories, and is used to gather test data.

PATS Physical Characteristics

The PATS hardware presently consists of twenty-two 6" x 9" boards that fit into a chassis that is slightly larger than an ATR box. Much of the space is used for spacing between cards because of sockets used in the hardware build. The system draws about 200 watts of DC power from the power supplies. The power supplies provided as part of the hardware will operate with either 60 Hz or 400 Hz, 115 volts AC line power.

SUMMARY

The PATS hardware is designed to reduce operator workload, and provide real time multiple recognition. It does not tire like human operator and hence will operate consistently and reduce response time.

With appropriate modification, the system can be used for target acquisition, weapon delivery and missile guidance.

REFERENCES

1. Soland, D. E., and P. M. Narendra, "Prototype Automatic Target Screener", Proceedings, SPIE, Volume 178, April, 1979.

2. Narendra, P. M., and R. C. Fitch, "Real Time Adaptive Contrast Enhancement for Imaging Sensors," Proceedings, SPIE, Volume 137, May, 1978.

The authors wish to thank Dr. P. Narendra, Mr. R. Larson, and Dr. D. Panda for initial algorithm simulation studies; Mr. H. Huber and Mr. S. Elvine for the building the hardware; and Ms. K. Huber for typing this manuscript.

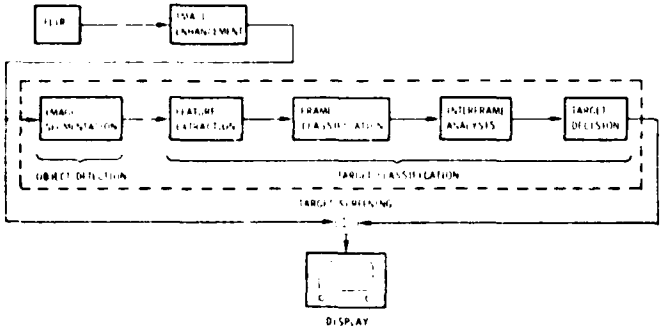


Figure 1. PATS Target Screening Function

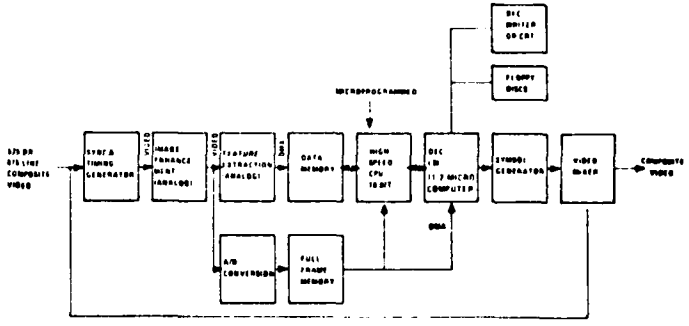


Figure 2. PATS Hardware

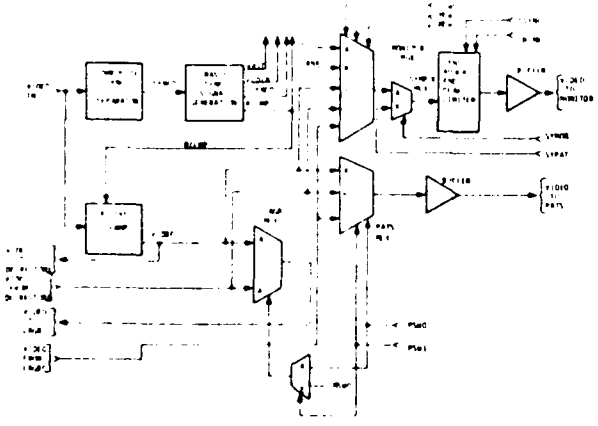


Figure 3. Sync Separation and Switching

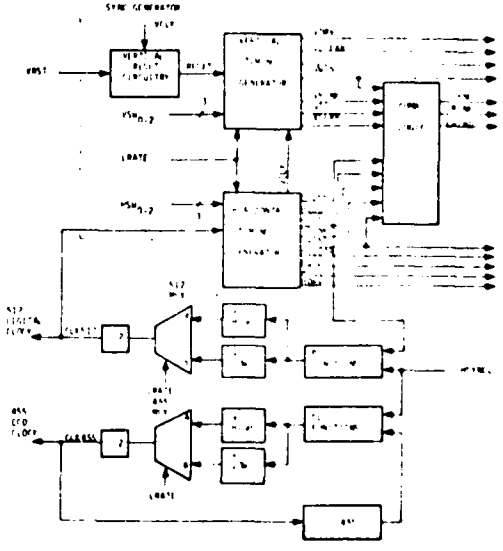


Figure 4. System Timing Generator

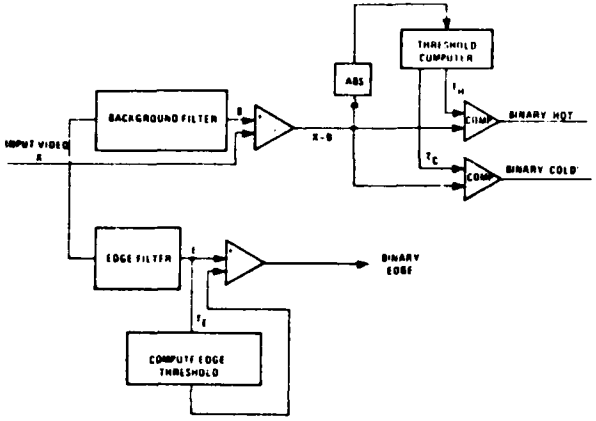


Figure 5. Autothreshold Hardware

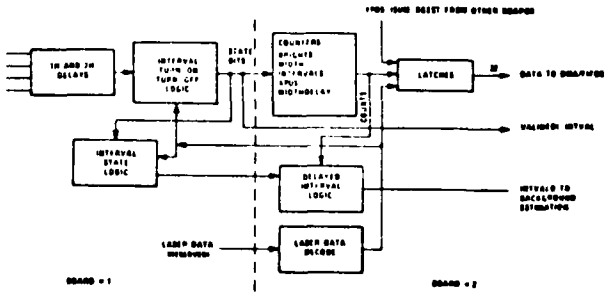


Figure 6. Interval Hardware

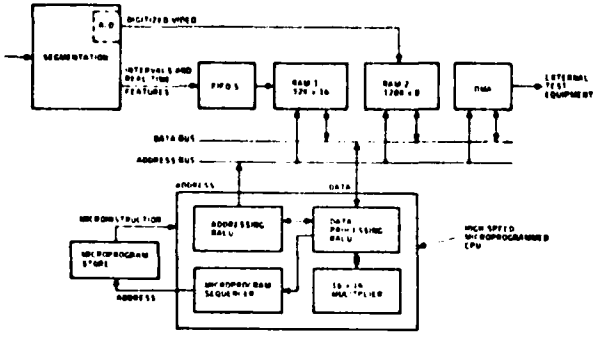


Figure 7. CPU1 System Architecture

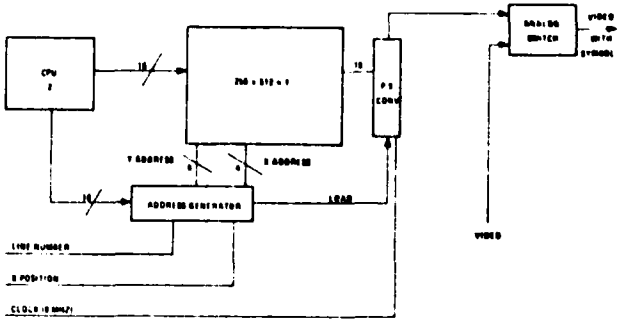


Figure 8. Symbol Generation

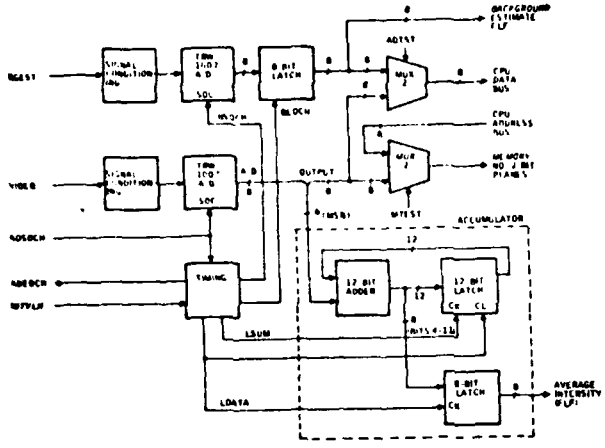


Figure 9. Analog-To-Digital Converter Unit

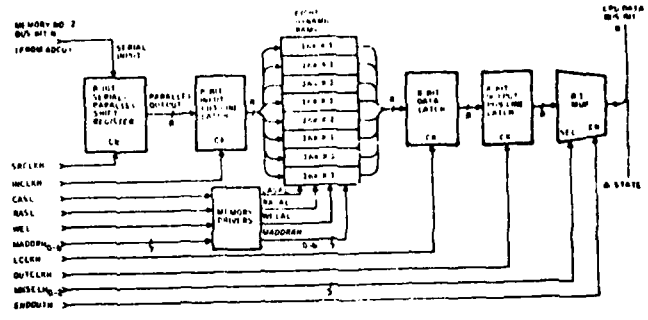


Figure 10. Memory Bit Plane

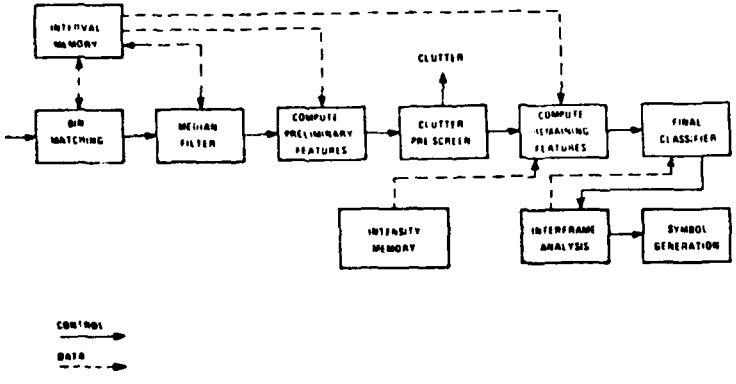


Figure 11. Software Functions

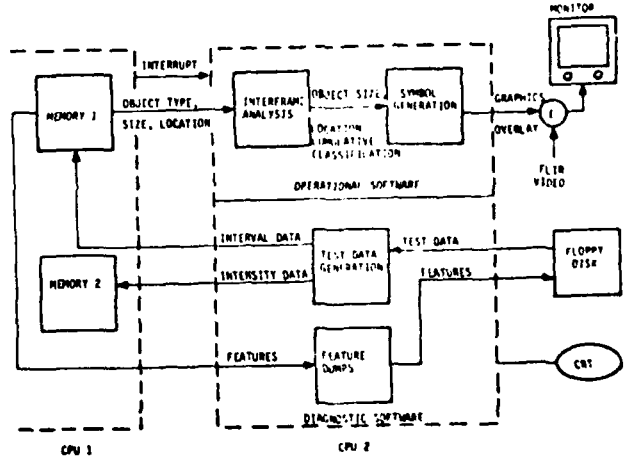
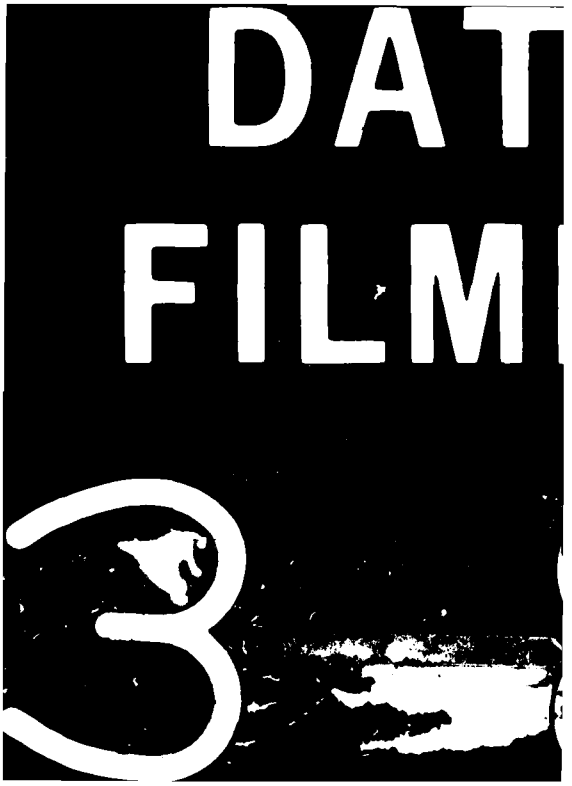


Figure 12. CPU2 Software Functions


DAT
FILM